An investigation into real time modelling techniques for power system fault transients

Fu, Wing Ming

Award date:
1992

Awarding institution:
University of Bath

AN INVESTIGATION INTO REAL TIME MODELLING TECHNIQUES FOR POWER SYSTEM FAULT TRANSIENTS

Submitted by
FU, Wing Ming, B.Sc. (Hons),
for the degree of
Ph.D. of the
University of Bath
1992

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Summary

This report is concerned with an investigation into new techniques for the real-time modelling of fault transients on EHV transmission systems during a fault generated autoreclosure sequence.

The report concentrates on three techniques, namely, time-domain convolutional, frequency-domain, and EMTP simulation techniques.

Based on the computational time measured, the feasibility of the respective technique for the purposes of developing a real time simulator in order to interactively test a modern high speed protection relay is discussed.

The simulation principles and techniques are described and sample outputs are compared with the results of the traditional mainframe simulator program.

The optimizations carried out to the simulation codes in order to obtain the best computational time are described in some detail.

Finally, future development of the project is discussed.

Acknowledgements

The author is indebted to his project supervisors, Mr. A.R. Daniels and Dr. R.K. Aggarwal, Senior Lecturers at the University of Bath, for their help and guidance throughout this work.

Financial support from Bath University's Research Bursary and Overseas Research Studentships is gratefully acknowledged.

The author would like to express his gratitude to Professor A.T. Johns and Dr. R.W. Dunn for their invaluable advice and help in the project.

He would like to thank Mr. V.S. Gott and B. Ross for assistance with the computing facilities. He would also like to express his thanks to his colleagues at the University for their advice and help.

Finally, the author would like to thank his parents and Miss Ming Mei Ang. Without their support and constant encouragement, this work would have never been completed.

Contents

List of principal symbols

General

$h(t)$	unit step function
j	complex operator
s	Laplace operator
t	time
τ	travelling time
Δt	time step length

System variables

I_s	sending end current
V_s	sending end voltage
E_s	sending end circuit breaker voltage
I_f	fault path current
V_f	fault point voltage
E_{ff}	fault point forcing voltage
E_r	receiving end circuit breaker voltage
I_r	receiving end current
V_r	receiving end voltage

Source parameters

Z_{ss}	sending end source impedance matrix
Z_{sr}	receiving end source impedance matrix

Line parameters

x	distance from sending end
l	line length between node k and node m
L	line length between sending and receiving end

R resistance per unit length
L inductance per unit length
C capacitance per unit length
G susceptance per unit length

A_1 ABCD matrices of the system model
B_1
C_1
D_1
A_f
B_f
C_f
D_f
A_2
B_2
C_2
D_2

Q current eigenvector matrix
S voltage eigenvector matrix
γ propagation constant matrix
ψ $S\gamma S^{-1}$

Z_f fault impedance matrix
Y_f fault admittance matrix
Z_c surge impedance matrix
Y_c surge admittance matrix
Z_{mc} modal surge impedance matrix
Y_{mc} modal surge admittance matrix

Machine variables

I_t machine terminal current
V_t machine terminal voltage
ω angular frequency
ω_s rated synchronous angular frequency
θ d-axis rotor position w.r.t. phase-a axis
δ synchronous torque angle

Machine parameters

R_a	armature resistance
X_l	armature leakage reactance
X_0	zero-sequence reactance
X_d, X_q	d-axis, q-axis reactances
X'_d, X'_q	d-axis, q-axis transient reactances
X''_d, X''_q	d-axis, q-axis subtransient reactances
τ'_{do}, τ'_{qo}	transient open-circuit time constants
τ''_{do}, τ''_{qo}	subtransient open-circuit time constants

H	inertia constant
J	moment of inertia

Fault sequences

t_{fault}	fault inception time
t_{open}	breaker pole opening time
t_{clear}	fault clearing time
$t_{reclose}$	breaker pole reclosing time

Modified Fourier transform

a	frequency shift constant
Ω	truncation frequency
$f(\omega - ja)$	modified Fourier transform
σ	sigma factor

Subscripts

a, b, c	phases a, b and c
d, q	direct-axis, quadrature-axis
f	fault point
s, r	sending end, receiving end
$t1$	fault inception stage
$t2$	breaker pole opening stage
$t3$	fault clearing stage
$t4$	breaker pole reclosing stage

Abbreviations

CDL	Component Distribution Language
EHV	Extra High Voltage
EMTP	ElectroMagnetic Transient Program
FDS	Frequency Domain Simulator
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
IFFT	Inverse Fast Fourier Transform
IIR	Infinite Impulse Response
RISC	Reduced Instruction Set Computers
s.c.l.	short circuit level

List of Figures

Chapter 1

Introduction

1.1 Real time simulation of power lines

Power system analysis generally falls into two broad categories of study: steady-state and fault transient. Much effort has been devoted in recent years to improving the techniques for steady-state analysis as the size and complexity of the power system networks increases. The fault transient study can be subdivided into electromagnetic and electromechanical. The former is mainly concerned with the study of the high frequency travelling waves on the power transmission line system. The latter is mainly concerned with the study of the low frequency transients caused, for example, by pole slipping of the power system generators, particularly during the period when the line breakers are open in an autoreclosure sequence.

In the application of autoreclosure techniques to a power transmission system, there are essentially two periods of interest: the prefault and postfault clearance periods. During the prefault clearance period, transients are mainly electromagnetic, while during the latter period, the transient phenomenon, particularly

during the so-called 'dead period', is primarily associated with lower frequencies, such as electromechanical.

Both the aforementioned fault transients produced during an autoreclosing sequence can cause a relay to malfunction, which can be detrimental to the reliability and stability of a transmission system as a whole. Hence, it is essential to model these transients correctly and to be able to test a modern protective relay interactively in real time during the full autoreclosure sequence.

1.1.1 Previous simulators

Modern high speed protection relays used with fast switch-gear operate such that the time between when a tripping signal is sent by a relay to open a circuit breaker to the time when the contacts of the circuit breaker start to separate, is typically around $40ms$. This means that for 'genuine' real time simulation, the simulator must be able to calculate all the circuit breaker opening transients within $40ms$ in order to test the relays interactively.

Whereas analogue simulators use scaled down models of system elements, a digital simulator models these same elements by solving the mathematical equations which describe the element behaviour. In order to run an electromagnetic transient study for a fault on a plain double end fed transmission system as shown in fig. 1.1, hundreds of millions of floating point operations per second (MFLOPS) are required to solve all the necessary equations in a single time step. It is clear that parallel processing techniques running on processing elements, capable of very high speed floating point arithmetic calculations, can be usefully employed. In other words, it would probably be impossible to achieve the 'real time' require-

ment if the simulator had been designed to run in a serial mode using a single processor.

Several authors [1, 2] have successfully developed simulators for testing modern high speed protective relays. These simulators have been designed to run in a serial mode, and can only test the relays in 'pseudo' real time. That is, they perform many sets of simulations off line to cover each eventuality so that when a real time test sequence is performed, the time at which the relay operates the circuit breakers determines which particular stored transient waveform is to be played back. This system has several disadvantages. Firstly, a large amount of memory is needed to store all the waveforms generated. Secondly, the effect of high frequency transients caused by fault clearance and autoreclosure cannot be studied as too many waveforms would have to be stored.

These simulators can only be satisfactorily used to test the relays up to prefault clearance period caused by line faults. They cannot be used to test the relays interactively in real time for the full fault autoreclosure sequence, i.e., during both the prefault and postfault clearance periods. During the latter period, particularly the period prior to circuit breaker reclosure, the low frequency electromechanical transients produced by, for example, the pole slippings of the power system generators may have a detrimental effect on the relays and therefore must be modelled.

1.1.2 The project

With the advent of modern digital computing technology, the low cost of microprocessor based computers has allowed them to be dedicated to a range of

tasks, particularly where immediate response is more important than a high peak throughput. Exploiting concurrency with multiprocessor systems can increase system throughput above that of the individual processing elements and makes it possible to realise the 'genuine' real time simulation.

The objectives of this project are to develop and investigate various approaches for the power system fault transient simulation, and based on the computational time measured, ascertain which of the simulation methods would be most suited to developing a simulator to test relays interactively, in real time, during both the prefault and postfault clearance periods.

Simulation models based on the time domain convolutional technique [3, 4], the frequency domain simulation technique [1, 2] and the EMTP simulation technique [5, 6] have been further developed and investigated in this project.

The whole research work has been carried out in three stages. During the first stage, the main thrust has been on developing and optimizing the electromagnetic transient simulation based on the time-domain convolutional technique. During the second stage, the electromagnetic transient modelling using the frequency-domain simulation technique has been developed. Optimizations are carried out to the frequency-domain simulation codes in order to achieve the best computational time. These optimizations include designing multi-processing algorithms running on the transputers and handcrafting the simulating codes so that the parallel processing capabilities of the Intel i860 microprocessor can be exploited. During the third stage, both electromagnetic and electromechanical transient modelling based on the EMTP simulation program is developed.

It must be emphasised that a frequency dependent line parameter model has

been used to simulate the electromagnetic transients in which the high frequency components are dominant. Furthermore, throughout all the sequences involving the electromagnetic transient simulation, the generators are represented by a constant lumped series subtransient reactance and the generator speeds are assumed to be constant. Whilst this assumption is justified for the relatively short time period of prefault clearance, it is not strictly valid in the so-called 'dead period' once the fault has been removed. Hence, it is necessary to incorporate actual generator models into the simulator in order to be able to model the low frequency effects associated with electromechanical transients.

In contrast to the traditional mainframe computer simulators, another objective of the project is to show the feasibility of implementing the aforementioned simulator using a commercial hardware such as IBM-PC/AT so that a production grade software package can be produced.

1.2 Transient calculation techniques

In transient studies there are numerous calculation methods available. Each method presents its own merits and the choice of which method should be used depends very much on the accuracy of the results required, the available means of tackling the problem, and the time taken to obtain the results. These methods can be classified mainly into three groups, namely, those using a scaled model of the actual system, mathematical methods and methods using a combination of analogue and digital computers. The first method uses a Transient Network Analyser (TNA) [7] where actual system parameters are scaled down. It presents some advantages in that the system response to any study can be obtained in a very short time and, once the model is established, various studies can be

done very easily. However, its major drawbacks are in that it cannot accurately represent frequency dependent parameters of the system or perform simultaneous scaling down of all system parameters. Furthermore, it requires a great deal of time and effort to re-model the system for different system parameters.

In recent years, however, significant progress has been made in power system transient studies using digital computers. Method developed by Bergeron [8], and those based on travelling wave [9, 10, 11], Fourier transform [1, 2, 12, 13, 14, 15], EMTP [5, 6] and time-convolution methods [4, 16, 17] for electromagnetic and electromechanical transient analyses in power systems are now widely used to provide practical test facilities comparable and in many cases, results are better than those attained from transient network analysers.

1.2.1 Time-domain convolutional technique

Following early work by Budner [3] and by Umoto and Hara [18], the common theme has been that of expressing nonlinear dependencies in the frequency-domain in impulse responses in the time-domain. Basically, the time convolutional technique refers specifically to the process by which numerical solutions for transmission line impulse response functions, as formed from inverse Fourier transform evaluations, are used in time-convolution. As shown later in Chapter 6, the simulation technique mathematically involves a large number of convolutional integral calculations. From the real time simulation point of view, the major disadvantage of this method is that the lengthy and time consuming convolutional integral calculations involved to achieve a reasonable degree of accuracy.

1.2.2 Frequency-domain technique

The frequency domain technique developed here is based on the previous work of Johns and Aggarwal [2]. In order to carry out frequency-domain calculations, two important routines are used, i.e., the Fast Fourier Transform (FFT) and the Inverse Fast Fourier Transform (IFFT). Firstly, all the time-domain data required for the transient calculations of a particular event is transformed into the frequency-domain using the FFT. After completing all the transient calculations, the resultant data is transformed back into the time-domain using the IFFT. For real time purposes, frequency spectra of the system parameters are pre-calculated and stored. Methods for calculating these parameters in multi-phase lines are based on techniques developed by Carson [19]. The major disadvantage of the frequency domain technique is that, since transient calculations have to be performed in the frequency domain, data has to be transformed into and out of the frequency domain every time a circuit change occurs in the time domain.

1.2.3 EMTP simulation technique

EMTP has been developed for simulating power system transients in the time domain. Basically, it synthesizes an electrical network by a set of system equations which are formed by using nodal circuit analyses and then solving all the equations numerically in the time domain. The EMTP method of frequency-dependent transmission line modelling can be summarized as follows:

- Calculation of desired transfer functions in the frequency domain.
- Numerical transformation of these functions into the time domain and synthesis of an equivalent network by direct fitting in the frequency domain.

- Calculation of transients in the time domain by solving equations of the synthesized network.

1.3 Simulation hardware and software

The simulator software was originally developed under the Helios Operating System [20], which was hosted in an IBM-PC with several T800 transputer plug-in boards [21]. In order to obtain benchmark calculation time for the simulators, the programs were then ported to an Intel 80386 [22] microprocessor based IBM-PC with an Intel i860 [23] microprocessor plug-in board. The 'Number-Smasher', i860 is well known for its fast speed in manipulating floating point operations. According to the data given by Fried [24], amongst all the other microprocessors, the i860 can achieve the highest number of MFLOPS operations. The EMTP programs were developed on an 80386 based IBM-PC with an Intel 80387 maths-coprocessor. Detailed discussions of simulation hardware and software are given in Chapter 7.

In general, the results obtained from the newly developed simulators have very close correspondence with those obtained from the well tried and tested frequency domain models developed by Johns and Aggarwal [1, 2] on the mainframe computers. Detailed simulation results and discussions are given in Chapter 8.

1.4 Summary of the thesis

Chapter 2 reviews the basic principles associated with autoreclosure schemes and shows the essential requirements for the successful implementation of a power

system fault transient simulator for testing a relay interactively in real time.

Chapter 3 derives the equations for analyzing a single conductor and a three phase transmission line system. A distributed line parameter model is used. Modal analysis for a three phase system is outlined.

Chapter 4 sets up the electromagnetic transient models based on the frequency domain formulation and various time domain formulations.

Chapter 5 sets up a detailed power system source model which is used for the electromechanical transient studies. The source model consists of an actual generator model, an AVR, an governor control system and an generator transformer.

Chapter 6 elaborates on the numerical calculation techniques for simulating various autoreclosure sequences based on the models derived in Chapter 4 and 5.

Chapter 7 describes the simulation hardware and software. The design of multiprocessing algorithms and the optimizations of the simulating codes are described. Handcoding using the Intel i860 microprocessor's parallel computing capabilities is also outlined in this chapter.

Chapter 8 validates and discusses the simulation results.

Chapter 9 draws the overall conclusions.

Chapter 10 presents the proposed future work.

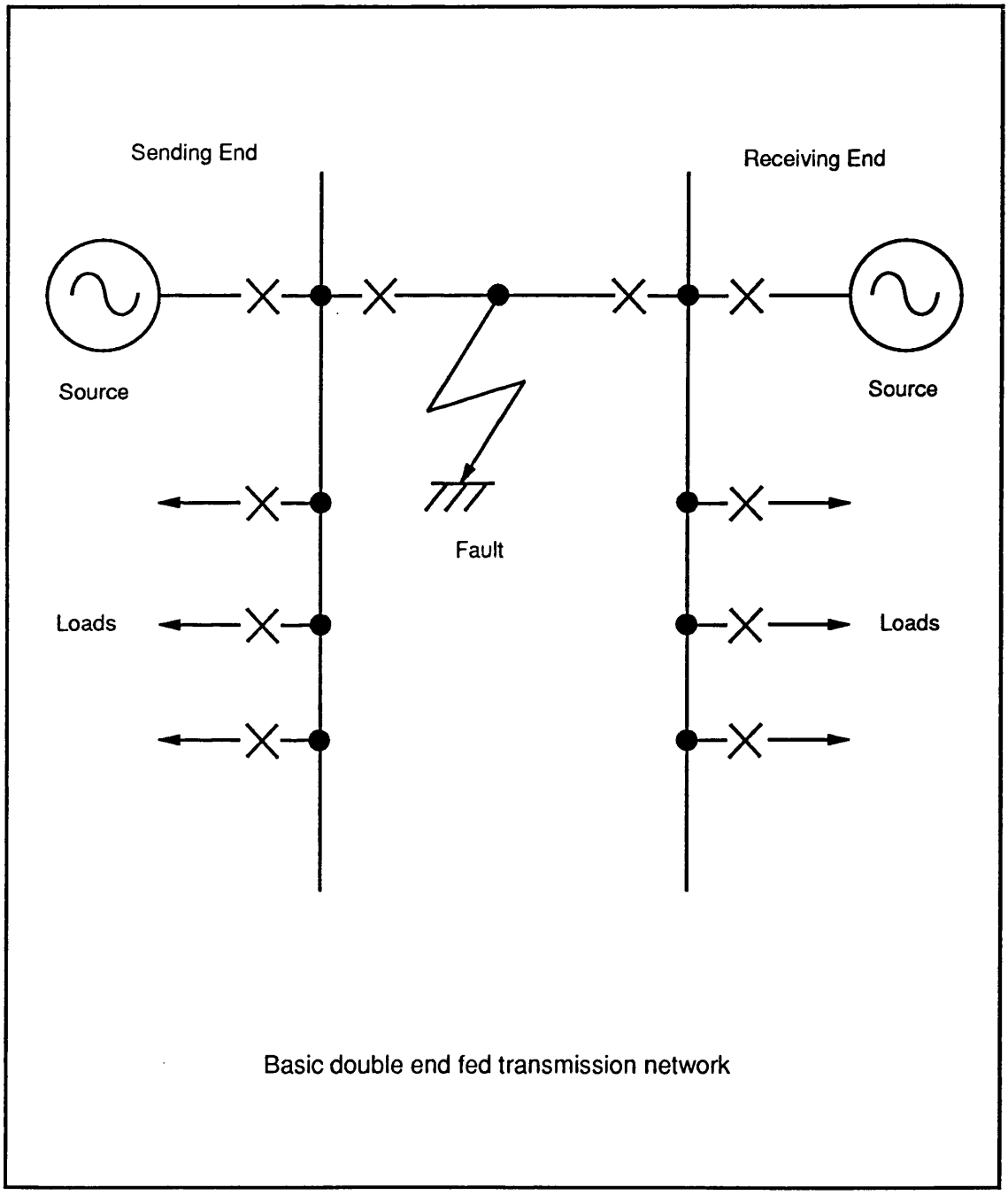

Figure 1.1: Basic 2-ended transmission system

Chapter 2

Autoreclosure schemes and real time requirements

2.1 Introduction

In the majority of transmission line fault incidents, if the faulty line is immediately tripped out, and time is allowed for the fault arc to de-ionize, reclosure of the circuit breakers will result in the line being successfully re-energized. Auto-reclose schemes are employed to carry out this duty automatically; they have been the cause of a substantial improvement in continuity of supply.

The successful operation of such a scheme requires that the protective relay concerned must be able to diagnose a fault condition and initiate the appropriate switching operations. The design and testing of the relay requires the knowledge of the corresponding voltage and current signals following any fault conditions. Since it is impractical to test a relay using an actual power system, particularly in terms of the very high costs involved, the development and testing of any new protective relays has to be dependent on accurate modelling of the fault

transients.

As the size and complexity of power systems increases, coupled with the advent of high speed microprocessor based protective relay and fast operating switchgear, a very severe burden is imposed on the real time simulator in terms of both power and speed. The simulator must be able to produce the correct information in a very short time period for each event of the fault sequence in order to be able to test the relay interactively in real time.

This chapter reviews the basic principles associated with autoreclosure schemes on EHV systems and shows the essential requirements for the realization of a power system fault transient simulator for real time relay testing.

2.2 Requirements for successful operations

Fig. 2.1 and fig. 2.2 show the sequence of events in a typical auto-reclose operation, where the circuit breaker makes one attempt at reclosure after tripping to clear a fault. Two conditions are illustrated. Fig. 2.1 shows a successful reclosure in the event of the fault being transient. Fig. 2.2 shows an unsuccessful reclosure followed by lock-out of the circuit breaker if the fault is permanent. Appendix A gives detailed definitions such as dead time, arcing time, de-ionizing, closing time, etc.

On EHV transmission lines, the most important consideration in the application of auto-reclosing is the maintenance of system stability and synchronism. For a double-end fed power system as shown in fig. 1.1, the successful application of auto-reclosing requires high speed protection, fast operating circuit breakers and

a short dead time.

The first requirement for the application of high speed auto-reclosing is a knowledge of the system disturbance time that can be tolerated without loss of system stability. The power/angle curve together with the value of electrical power transmitted, allows the permissible load angle change to be estimated. It is then necessary to have some knowledge of the load angle change/time relationship in order to estimate the maximum permissible system disturbance time. A knowledge of protection and circuit breaker operating characteristics and fault arc de-ionization times is hence necessary to assess the feasibility of high speed auto-reclosing in any given system.

1. **Protection characteristics**: The use of high speed protective gear, such as distance or pilot wire schemes, giving operating times of the order of $30ms$ or less, is essential. In conjunction with fast operating circuit breakers, high speed protection reduces the duration of the fault arc and thus the total system disturbance time. If single-phase tripping and auto-reclosing is used, especially on long transmission lines, the faulty phase should be disconnected for a longer time interval than in the case of three-phase tripping and auto-reclosing, in order to obtain an equal probability of successful reclosure. The reason for this is that, during the period when the faulty phase is open, capacitive coupling between the healthy phases tends to maintain the arc.

2. **Circuit breaker characteristics**: With a large concentration of electrical power and high fault levels involved on EHV systems, the high speed auto-reclose cycle imposes a very severe duty on the performance of circuit breakers. The accepted breaker cycle of break-make-break requires the circuit breaker to interrupt the fault current, reclose the circuit after a time

delay of $0.2s$ upwards and then break the fault current again if the fault persists. Fig. 2.3 shows a trip-close operation for a pneumatically operated $132kV$ breaker, showing the dead time which can be attained with and without the special high speed reclosing mechanism. Fig. 2.4 shows the operating times for typical $400kV$ pressurized head air blast breakers.

3. **Choice of dead time**: The dead time setting on a high speed auto-reclose relay should be long enough to ensure complete de-ionization of the arc. On EHV systems, an unsuccessful reclosure is more detrimental to the system than no reclosure at all.

4. **Number of shots**: High speed auto-reclosing on EHV systems is invariably single shot. Repeated reclosure attempts with high fault levels would have serious effects on system stability, so the circuit breakers are locked out after one unsuccessful attempt.

2.3 Single and three phase autoreclosure

When three-phase auto-reclosing is applied to single circuit interconnectors between two power systems, the tripping of all three phases on a fault causes an immediate drift apart of the two systems in their phase relation to one another. No interchange of synchronizing power can take place during the dead time. If, on the other hand, only the faulty phase is tripped during earth fault conditions, which account for the vast majority of faults on overhead lines, synchronizing power can still be interchanged through the healthy phases.

For single-phase auto-reclosing, each phase of the circuit breaker must be segregated and provided with its own closing and tripping mechanism; this is normal

with EHV air blast breakers and most oil breakers. The associated tripping and reclosing circuitry is therefore complicated, and, except in distance schemes, the protection needs the addition of phase selecting relays. It is normal practice to arrange that, in the event of multi-phase faults, all three phases of the circuit breakers are tripped and locked out.

The advantages of single-phase auto-reclosing are the maintenance of transfer of synchronizing power, and, on multiple earth systems, negligible interference with the transmission of load. This is because the open phase current can flow through earth via the various earthing points until the fault is cleared and the faulty phase restored. The main disadvantage is the longer de-ionization time resulting from capacitive coupling between the faulty and healthy lines, which can cause interference with communication circuits, and in certain cases, mal-operation of earth fault relays on double circuit lines owing to the flow of zero sequence currents. These are induced by mutual induction between faulty and healthy lines.

2.4 Requirements for real time simulator

Referring to the timings of a typical autoreclosure sequence as shown in fig. 2.4, it can be seen that from the time that a tripping signal is sent by the relay to open a circuit breaker to the time that the breaker arc fully extinguished is $45ms$. This means that there is $45ms$ available for the simulator. Within this $45ms$, the simulator must be able to do two things. Firstly, from the fault inception transient waveforms, which have been calculated off-line, it must be able to locate the exact times of the current zero crossings of the respective phases. It is at these times that the circuit breaker poles of the respective phases are to be opened. Secondly,

it must be able to calculate all the circuit breaker pole opening transients.

Since each breaker pole opens only at the current zero of the respective phase, the breaker poles at both ends of a double end fed system *do not* operate simultaneously. There is a time stagger between the pole openings of different circuit breakers. For the worst case of a three phase autoreclosure operation of a double end fed single circuit system, the time interval between the opening of the first breaker pole and the last breaker pole may be less than $5ms$. Within this $5ms$, there will be 5 subsequent circuit changes produced by the five remaining breaker pole openings and this means that, for the worst case, the simulator must be able to produce five different sets of transients within $5ms$, giving an average of $1ms$ for the simulator to produce each set of pole opening transients.

The aforementioned worst case scenario highlights the computational requirements necessary for implementing a truely real time power system simulator for testing protective relays. It should be mentioned that the work presented here investigates, and to a certain extent attempts to optimize, the various options available for achieving such a real time simulator. Furthermore, it also makes suggestions as to what sort of state of the art hardware could possibly be designed and engineered to meet these requirements, thus laying the foundation for the development of an interactive real time relay testing facility in the future.

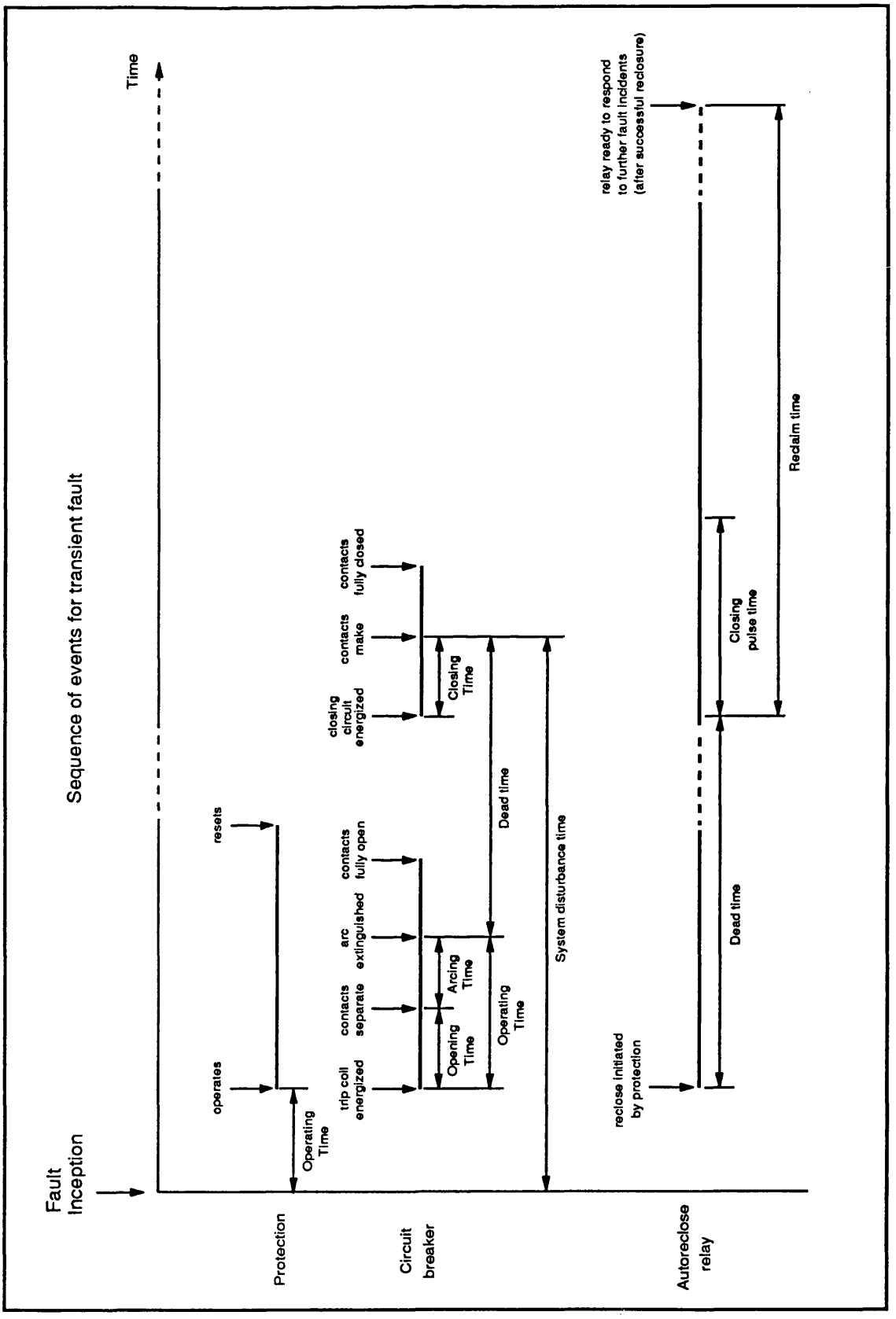

Figure 2.1: Autoreclose sequence for a transient fault.

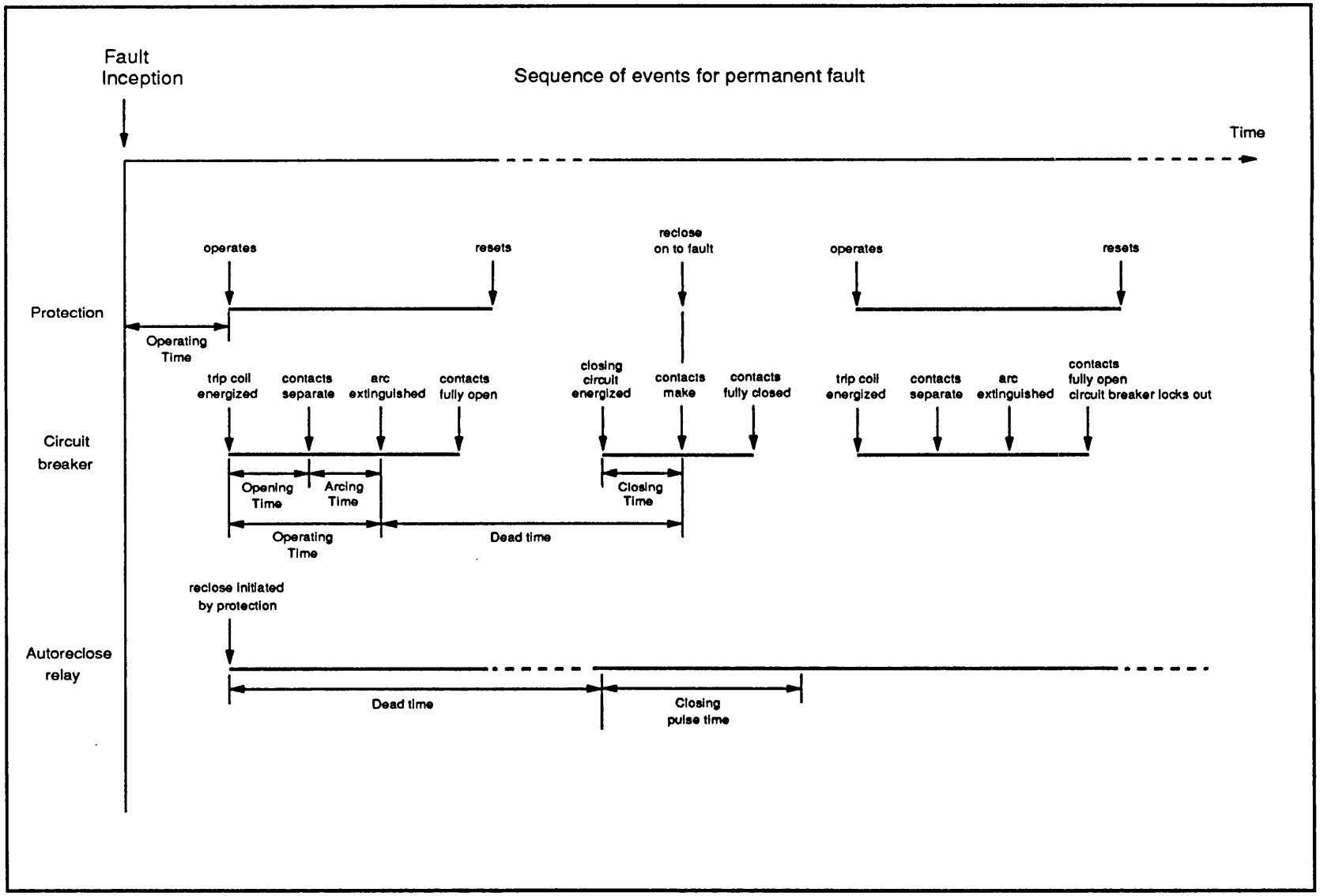

Figure 2.2: Autoreclose sequence for a permanent fault

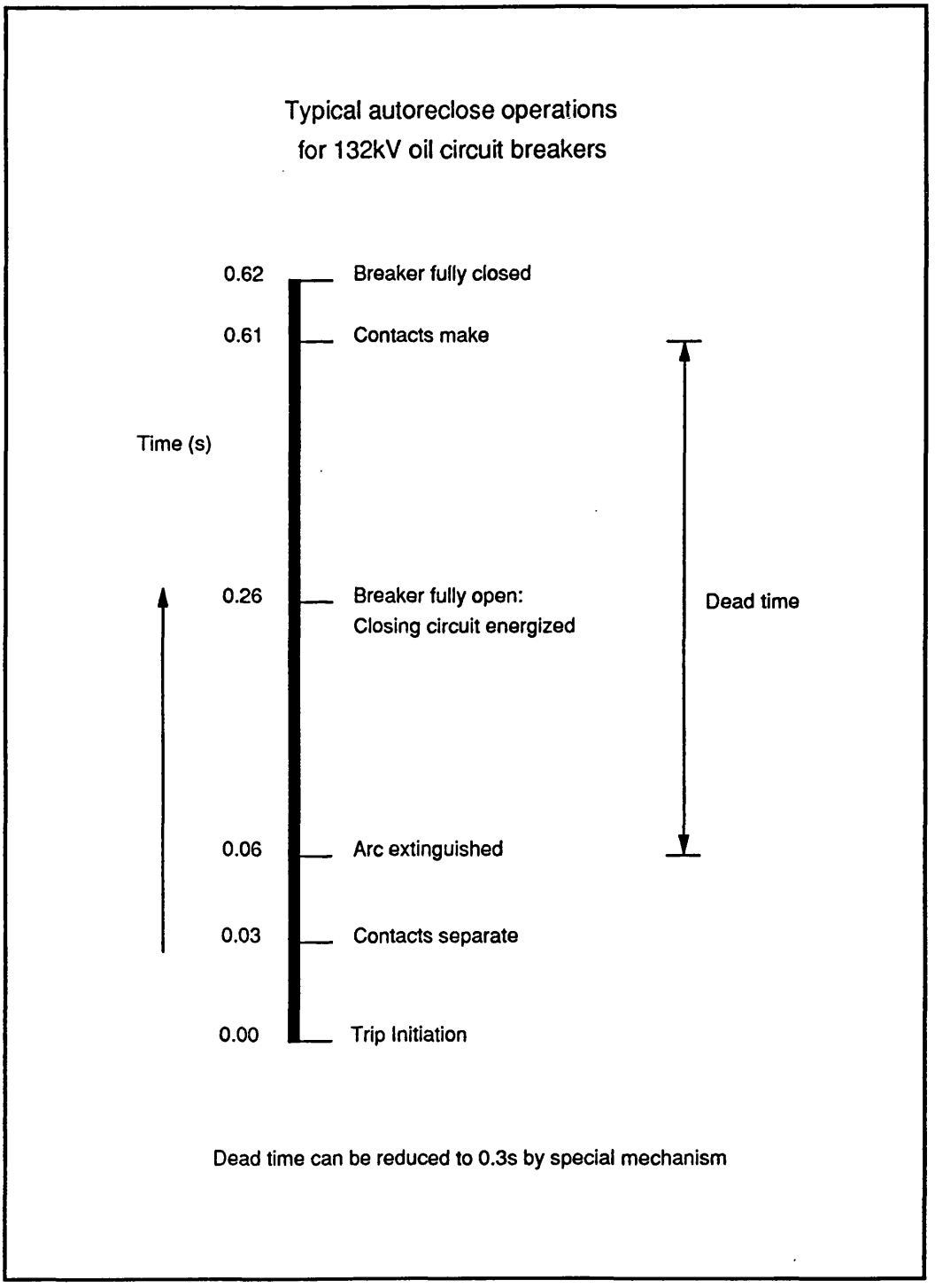

Figure 2.3: Typical autoreclose operations for 132kV oil circuit breakers

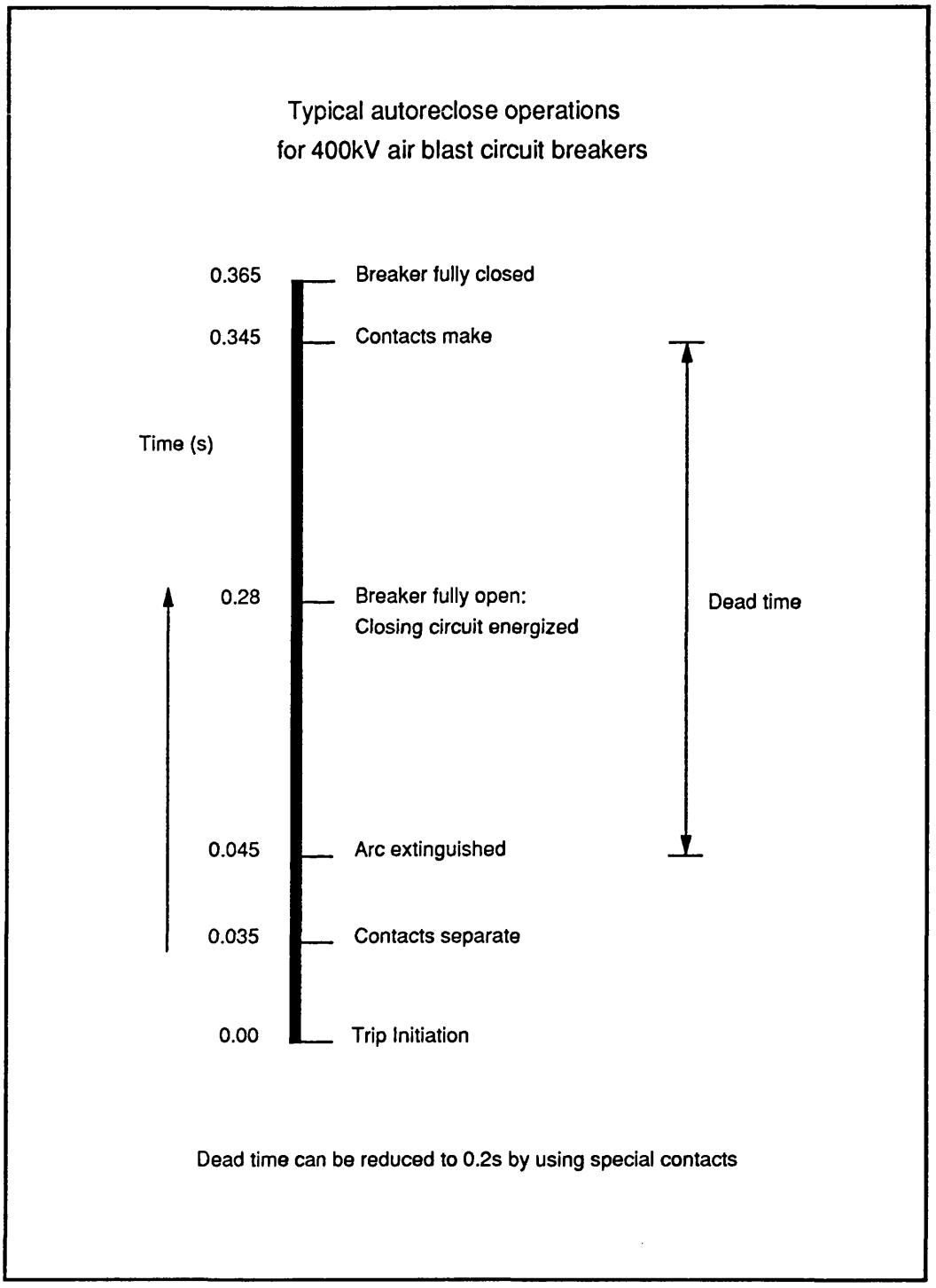

Figure 2.4: Typical autoreclose operations for 400kV air blast circuit breakers

Chapter 3

Transmission line theory

3.1 Introduction

This chapter establishes the mathematical formulations for a single phase and a three phase transmission line system. Line models with distributed parameters are used. Modal analysis is employed so that a three phase system is decoupled into three separate single phases and treated as individual single phase problem. The frequency dependence of the line parameters is illustrated.

3.2 Single conductor line

The series resistance and inductance and the leakage conductance and capacitance to earth of a transmission line are distributed over the entire length of the line. This can be represented as shown in fig. 3.1. Following some circuit analyses, the following transmission line equations are obtained:

$$-\frac{dV_x}{dx} = (R + j\omega L)I_x \tag{3.1}$$

$$-\frac{dI_x}{dx} = (G + j\omega C)V_x \tag{3.2}$$

where x is the distance from the sending end to the point of interest. R, L, G and C are respectively resistance, inductance, leakage condutance and capacitance per unit length of the line. By differentiating the first equation with respect to x, and replacing the current derivative with the second equation, a second order differential equation for voltages only is obtained:

$$\frac{d^2V_x}{dx^2} = \gamma^2 V_x \tag{3.3}$$

where $\gamma = \sqrt{(R + j\omega L)(G + j\omega C)}$.

Similarly, a second order differential equation for currents only can be obtained:

$$\frac{d^2I_x}{dx^2} = \gamma^2 I_x \tag{3.4}$$

The general solution for the above equations in terms of the sending end voltage V_s and sending end current I_s are given as shown below:

$$\begin{bmatrix} V_x \\ I_x \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_s \\ I_s \end{bmatrix} \tag{3.5}$$

where

$$\begin{aligned} A &= \cosh(\gamma x) \\ B &= Z_c \sinh(\gamma x) \\ C &= \sinh(\gamma x)/Z_c \\ D &= \cosh(\gamma x) \\ Z_c &= \sqrt{\frac{R + j\omega L}{G + j\omega C}} \end{aligned}$$

Z_c is defined as the characteristic impedance of the line.

3.3 Three phase line

An elemental section of a distributed single circuit mutually coupled three phase power transmission line is shown in fig. 3.2. Following similar mathematical manipulations as before, the following system equations can be obtained:

$$-\frac{dV_{phase}}{dx} = Z_{phase}I_{phase} \tag{3.6}$$

$$-\frac{dI_{phase}}{dx} = Y_{phase}V_{phase} \tag{3.7}$$

$$\frac{d^2V_{phase}}{dx^2} = Z_{phase}Y_{phase}V_{phase} \tag{3.8}$$

$$\frac{d^2I_{phase}}{dx^2} = Y_{phase}Z_{phase}I_{phase} \tag{3.9}$$

where Z_{phase} and Y_{phase} are 3×3 matrices and, V_{phase} and I_{phase} are 3×1 column vectors. It should be noted that the series and shunt parameter matrices Z_{phase} and Y_{phase} are given in per unit length of the line. They are evaluated to take into account the presence of any overhead earthwires and also to include the effect of the frequency variation of both resistive and inductive line and earth parameters. Since the network is homogeneous, both Z_{phase} and Y_{phase} are symmetrical and can be expressed as follows:

$$[Z_{phase}] = \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{21} & Z_{22} & Z_{23} \\ Z_{31} & Z_{32} & Z_{33} \end{bmatrix} \tag{3.10}$$

$$[Y_{phase}] = \begin{bmatrix} Y_{11}+Y_{12}+Y_{13} & -Y_{12} & -Y_{13} \\ -Y_{21} & Y_{21}+Y_{22}+Y_{23} & -Y_{23} \\ -Y_{31} & -Y_{32} & Y_{31}+Y_{32}+Y_{33} \end{bmatrix} \tag{3.11}$$

The line parameters in the above matrices are defined as illustrated in fig. 3.2. It should be noticed that over the whole frequency range, the relationships of the preceding equations are nonlinear. But at any one frequency, they are linear. The matrix products $Z_{phase}Y_{phase}$ in eqn. 3.8 are in reverse order from that of $Y_{phase}Z_{phase}$ in eqn. 3.9 and are therefore different.

3.3.1 Modal analysis

It is not an easy task to find the solutions for the above second order differential equations directly. The solutions become simpler if the coupled equations can be transformed into decoupled equations. That is, if each phase is treated and solved as an individual single phase problem.

With eigenvalue theory, it becomes possible to transform the two coupled equations 3.8 and 3.9 from phase quantities to 'modal' quantities in such a way that the equations become decoupled, or in terms of matrix algebra, that the associated matrices become diagonal, e.g., for the voltages:

$$\frac{d^2V_{mode}}{dx^2} = \Lambda V_{mode} \tag{3.12}$$

where Λ is a diagonal matrix. To get from eqn. 3.8 to eqn. 3.12, the phase voltages must be transformed to modal voltages, i.e.,

$$V_{phase} = SV_{mode} \tag{3.13}$$

and

$$V_{mode} = S^{-1}V_{phase} \tag{3.14}$$

Thus, eqn. 3.8 becomes

$$\frac{d^2V_{mode}}{dx^2} = S^{-1}Z_{phase}Y_{phase}SV_{mode} \tag{3.15}$$

which, when compared with eqn. 3.12, shows that

$$\Lambda = S^{-1}Z_{phase}Y_{phase}S \tag{3.16}$$

To find the matrix S which diagonalizes $Z_{phase}Y_{phase}$ is the eigenvalue/eigenvector problem. The diagonal elements of Λ are the eigenvalues of the matrix product

$Z_{phase}Y_{phase}$ and, S is the matrix of eigenvectors or modal matrix of that matrix product. Assuming that the diagonal matrix Λ has the following form:

$$[\Lambda] = \begin{bmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_2^2 & 0 \\ 0 & 0 & \lambda_3^2 \end{bmatrix} \tag{3.17}$$

where λ_1^2, λ_2^2 and λ_3^2 are the corresponding eigenvalues, and the following equations can be written:

$$\frac{d^2V_{m1}}{dx^2} = \lambda_1^2 V_{m1} \tag{3.18}$$

$$\frac{d^2V_{m2}}{dx^2} = \lambda_2^2 V_{m2} \tag{3.19}$$

$$\frac{d^2V_{m3}}{dx^2} = \lambda_3^2 V_{m3} \tag{3.20}$$

The components, V_{m1}, V_{m2} and V_{m3} are associated with *Natural Modes of Wave Propagation.* Hence, the transformation matrix S is called the *Modal Voltage Matrix.* Each of the preceding equations can thus be manipulated as an ordinary differential equation.

Similarly, transformation can be performed for the current, i.e.,

$$I_{phase} = QI_{mode} \tag{3.21}$$

$$I_{mode} = Q^{-1}I_{phase} \tag{3.22}$$

where Q is the current transformation matrix. Although the matrix products in eqn. 3.8 and eqn. 3.9 have different eigenvectors, their eigenvalues are identical. While Q and S are different from each other, both can be related to each other as:

$$Q = [S^{\,t}]^{-1} \tag{3.23}$$

where 't' indicates transposition. Similarly, eqn. 3.9 can be transformed to

$$\frac{d^2I_{mode}}{dx^2} = \Lambda I_{mode} \tag{3.24}$$

with the same diagonal matrix as in eqn. 3.12. Thus, the following decoupled equations can also be written for the currents:

$$\frac{d^2 I_{m1}}{dx^2} = \lambda_1^2 I_{m1} \tag{3.25}$$

$$\frac{d^2 I_{m2}}{dx^2} = \lambda_2^2 I_{m2} \tag{3.26}$$

$$\frac{d^2 I_{m3}}{dx^2} = \lambda_3^2 I_{m3} \tag{3.27}$$

Fig. 3.3 schematically illustrates the physical interpretation of modal analysis for a three phase system.

3.3.2 Significance of S matrix

Assuming that the line is transposed, the S matrix will be (a): frequency independent, i.e., constant, and, (b): have scalar values. Using Clerk's transformation matrices, in which, assuming ideally transposed line, S and Q have the following values:

$$[S] = [Q] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & -2 \\ 1 & -1 & 1 \end{bmatrix} \tag{3.28}$$

$$[S^{-1}] = [Q^{-1}] = \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 3/2 & 0 & -3/2 \\ 1/2 & 1 & 1/2 \end{bmatrix} \tag{3.29}$$

Thus, for any point on the line, the voltages and currents are given by:

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & -2 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} V_{m1} \\ V_{m2} \\ V_{m3} \end{bmatrix} \tag{3.30}$$

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & -2 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} I_{m1} \\ I_{m2} \\ I_{m3} \end{bmatrix} \tag{3.31}$$

According to the preceding two equations, the following three cases can be deduced:

1. Suppose that only mode-1 currents are present, then $I_a = I_b = I_c = I_{m1}$. This means that for mode-1 currents, the currents in each of the a-phase, the b-phase and the c-phase are equal and flow in the same direction. This also shows mode-1 currents can only return through the earth plane and therefore are often called *Earth Mode* components, which is analogous to zero sequence currents.

2. Suppose that only mode-2 currents are present, then $I_a = -I_c = I_{m2}$ and $I_b = 0$. Thus, for mode-2 currents:

   - The currents in the a-phase and in the c-phase have equal magnitudes, but flow in opposite directions.
   - The b-phase current is zero.

   This distribution of currents shows that mode-2 is an aerial mode of propagation.

3. Similarly, suppose that only mode-3 currents are present, then $I_a = I_c = I_{m3}$ and $I_b = -2I_{m3}$. This distribution of currents also shows that mode-3 is an aerial mode of propagation.

3.3.3 Line equations in terms of modal components

With the decoupled equations 3.12 and 3.24 in modal quantities, each mode can be analysed as if it were a single conductor line. Comparing the modal equation,

$$\frac{d^2V_{mk}}{dx^2} = \lambda_k^2 V_{mk} \tag{3.32}$$

with the well-known equation of a single conductor line equation 3.3 as derived in section 3.2, i.e.,

$$\frac{d^2V_x}{dx^2} = \gamma^2 V_x$$

shows that the modal propagation constant γ_{mk} is the square root of the eigenvalue,

$$\gamma_{mk} = \alpha_k + j\beta_k = \sqrt{\lambda_k^2} = \lambda_k \tag{3.33}$$

where

α_k = attenuation constant of mode k, (e.g., Np/km)
β_k = phase constant of mode k, (e.g., in rad/km)

The phase velocity of mode k is defined as:

$$\text{phase velocity} = \omega/\beta_k \tag{3.34}$$

and the wavelength as:

$$\text{wavelength} = 2\pi/\beta_k \tag{3.35}$$

While the modal propagation constant is always uniquely defined, the modal series impedance and shunt admittance as well as the modal characteristic impedance are not, because of the ambiguity in the eigenvectors. Therefore, modal impedances and admittances only make sense if they are specified together with the eigenvectors used in their calculation. In order to find them, eqn. 3.6 must be transformed to modal quantities:

$$-\frac{dV_{mode}}{dx} = S^{-1} Z_{phase} Q I_{mode} \tag{3.36}$$

The triple matrix product in eqn. 3.36 is diagonal and, the modal series impedances are the diagonal elements of this matrix,

$$Z_m = S^{-1} Z_{phase} Q = Q^{\,t}\, Z_{phase} Q \tag{3.37}$$

Similarly, eqn. 3.7 can be transformed to modal quantities and, the modal shunt admittances are then the diagonal elements of the matrix,

$$Y_m = Q^{-1} Y_{phase} S = S^{\,t}\, Y_{phase} S \tag{3.38}$$

The proof that both Z_m and Y_m are diagonal is given by Wedepohl [25].

3.3.4 Approximate theory of wave propagation

As shown in the previous section, the modal propagation coefficients are complex functions evaluated from S and Q. But if the lines are ideally transposed, then $S = Q$ and, the value of γ becomes simply related to the basic line parameters by:

$$\gamma_1 = \sqrt{Z_0 Y_0} \tag{3.39}$$

$$\gamma_2 = \gamma_3 = \sqrt{Z_1 Y_1} \tag{3.40}$$

where Z_0, Y_0 are the zero sequence parameters per unit length of the line and, Z_1, Y_1 are the positive sequence parameters per unit length of the line. The modal surge impedances are given by:

$$Z_{c1} = \sqrt{\frac{Z_0}{Y_0}} \tag{3.41}$$

$$Z_{c2} = Z_{c3} = \sqrt{\frac{Z_1}{Y_1}} \tag{3.42}$$

Assuming that the transformation matrices S and Q are frequency invariant and $S = Q$, then the modal voltages and currents can be related directly to the actual phase values in the time domain using the following relationships:

$$V(t) = SV_{mode}(t) \tag{3.43}$$

$$I(t) = SI_{mode}(t) \tag{3.44}$$

The above procedure is exact in the case of balanced line configurations and still very accurate for transposed lines. In the more general case of unbalanced, untransposed lines, however, the modal transformation matrices are frequency dependent. Nevertheless, as concluded by Magnusson [26] and Wasley [27], it is still possible in this case to obtain a reasonably good approximation under the assumption of constant transformation matrices.

3.3.5 General solutions of modal equations

Now, consider the decoupled voltage and current equations in terms of modal propagation constants:

$$\frac{d^2V_{m1}}{dx^2} = \gamma_1^2 V_{m1} \tag{3.45}$$

$$\frac{d^2I_{m1}}{dx^2} = \gamma_1^2 I_{m1} \tag{3.46}$$

The general solutions of the preceding two equations are given by:

$$V_{m1} = X_1 e^{-\gamma_1 x} + X_2 e^{\gamma_1 x} \tag{3.47}$$

$$I_{m1} = Y_1 e^{-\gamma_1 x} + Y_2 e^{\gamma_1 x} \tag{3.48}$$

where X_1, X_2, Y_1 and Y_2 are arbitrary constants and are determined from the boundary conditions for a particular system. Thus, it follows from eqn. 3.47 that:

$$\frac{dV_{mode}}{dx} = \gamma \left[-X_1 e^{-\gamma x} + X_2 e^{\gamma x}\right] \tag{3.49}$$

where

$$\gamma = \begin{bmatrix} \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix} \tag{3.50}$$

Now, recall from eqn. 3.6 that,

$$-\frac{dV_{phase}}{dx} = Z_{phase} I_{phase} = Z_{phase} Q I_{mode} \tag{3.51}$$

$$-\frac{dSV_{mode}}{dx} = Z_{phase} Q I_{mode} \tag{3.52}$$

$$-\frac{dV_{mode}}{dx} = S^{-1} Z_{phase} Q I_{mode} \tag{3.53}$$

It can be shown that $S^{-1}Z_{phase}Q$ is a diagonal matrix [25] and it therefore follows that each modal voltage component is a function of the corresponding modal current component. It should be noted that the values γ_1, γ_2 and γ_3 are really *propagation constants* which define the characteristics of each mode. They are

usually called *Modal Propagation* constants. Combining eqns. 3.49 and 3.53, the following equation is obtained:

$$\begin{aligned} \frac{dV_{mode}}{dx} &= -S^{-1}Z_{phase}QI_{mode} \\ &= \gamma\left[-X_1e^{-\gamma x} + X_2e^{\gamma x}\right] \end{aligned} \tag{3.54}$$

Substituting the above equation with:

$$I_{mode} = Y_1e^{-\gamma x} + Y_2e^{\gamma x} \tag{3.55}$$

gives:

$$\begin{aligned} X_1e^{-\gamma x} - X_2e^{\gamma x} &= \gamma^{-1}S^{-1}Z_{phase}QI_{mode} \\ &= Z_{mc}\left[Y_1e^{-\gamma x} + Y_2e^{\gamma x}\right] \end{aligned} \tag{3.56}$$

where

$$Z_{mc} = \gamma^{-1}S^{-1}Z_{phase}Q \tag{3.57}$$

Z_{mc} is the modal surge impedance matrix and can be shown to be diagonal. Since Z_{mc} is diagonal, the modal voltages are only functions of the corresponding modal currents, i.e., mode-1 voltage is associated only with mode-1 current, etc. Since eqn. 3.56 must be true for all γ, it follows that:

$$X_1 = Z_{mc}Y_1 \tag{3.58}$$

$$X_2 = -Z_{mc}Y_2 \tag{3.59}$$

3.3.6 Derivations of ABCD matrices

Referring to fig. 3.2, assuming that the line length is L, the voltages and currents at the sending end ($x = 0$) are given by:

$$V_s = SV_{mode}\,|_{(x=0)} = S\left[X_1 + X_2\right] \tag{3.60}$$

$$I_s = QI_{mode}\,|_{(x=0)} = Q\left[Y_1 + Y_2\right] \tag{3.61}$$

and the voltages and currents at the receiving end ($x = L$) are given by:

$$V_r = SV_{mode}\,|_{(x=L)} = S\left[X_1 e^{-\gamma L} + X_2 e^{\gamma L}\right] \tag{3.62}$$

$$I_r = QI_{mode}\,|_{(x=L)} = Q\left[Y_1 e^{-\gamma L} + Y_2 e^{\gamma L}\right] \tag{3.63}$$

Thus,

$$V_s = SZ_{mc}\left[Y_1 - Y_2\right] \tag{3.64}$$

$$\left[Y_1 - Y_2\right] = Z_{mc}^{-1} S^{-1} V_s \tag{3.65}$$

Eqn. 3.61 gives:

$$\left[Y_1 + Y_2\right] = Q^{-1} I_s \tag{3.66}$$

Eqns. 3.65 and 3.66 can thus be solved for Y_1 and Y_2:

$$Y_1 = \frac{1}{2}\left[Z_{mc}^{-1} S^{-1} V_s + Q^{-1} I_s\right] \tag{3.67}$$

$$Y_2 = \frac{1}{2}\left[Q^{-1} I_s - Z_{mc}^{-1} S^{-1} V_s\right] \tag{3.68}$$

Substituting eqns. 3.67 and 3.68 into eqn. 3.63 gives:

$$I_{mr} = \cosh(\gamma L) I_{ms} - \sinh(\gamma L) Z_{mc}^{-1} V_{ms} \tag{3.69}$$

where

I_{mr} is the modal currents at the receiving end,
I_{ms} is the modal currents at the sending end and
V_{ms} is the modal voltages at the sending end.

Substituting eqns. 3.67 and 3.68 into eqn. 3.62 gives:

$$V_{mr} = \cosh(\gamma L) V_{ms} - \sinh(\gamma L) Z_{mc} I_{ms} \tag{3.70}$$

where V_{mr} is the modal voltages at the receiving end.

Rearranging eqns. 3.69 and 3.70 into matrix form, the following two-port matrix equation is obtained:

$$\begin{bmatrix} V_{ms} \\ I_{ms} \end{bmatrix} = \begin{bmatrix} \cosh(\gamma L) & Z_{mc}\sinh(\gamma L) \\ Y_{mc}\sinh(\gamma L) & \cosh(\gamma L) \end{bmatrix} \begin{bmatrix} V_{mr} \\ I_{mr} \end{bmatrix} \tag{3.71}$$

where $Y_{mc} = Z_{mc}^{-1}$.

Now using the following identities:

$$\begin{aligned} V_r &= SV_{mr} \\ I_r &= QI_{mr} \\ V_s &= SV_{ms} \\ I_s &= QI_{ms} \\ Z_{mc} &= \gamma^{-1}S^{-1}Z_{phase}Q \end{aligned}$$

eqn. 3.71 can be transformed back to phase quantities:

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} S\cosh(\gamma L)S^{-1} & Z_cQ\sinh(\gamma L)Q^{-1} \\ Y_cQ\sinh(\gamma L)S^{-1} & Q\cosh(\gamma L)Q^{-1} \end{bmatrix} \begin{bmatrix} V_r \\ I_r \end{bmatrix} \tag{3.72}$$

where

$$Z_c = SZ_{mc}Q^{-1} = \text{polyphase surge impedance} \tag{3.73}$$

$$Y_c = QY_{mc}Q^{-1} = \text{polyphase surge admittance} \tag{3.74}$$

Substituting eqn. 3.57 for Z_{mc}, the polyphase surge impedance matrix is then given by [25]:

$$Z_c = S\gamma^{-1}S^{-1}Z_{phase} \tag{3.75}$$

and

$$\psi = S\gamma S^{-1} \tag{3.76}$$

$$\cosh(\psi L) = S\cosh(\gamma L)S^{-1} \tag{3.77}$$

In this case, eqn. 3.72 can be expressed as shown below [2]:

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_r \\ I_r \end{bmatrix} \tag{3.78}$$

where

$$\begin{aligned} A &= \cosh(\psi L) \\ B &= \sinh(\psi L)Z_c \\ C &= Y_cBY_c \\ D &= Y_cAZ_c \end{aligned}$$

3.4 Frequency dependent parameters

The two most important parameters for wave propagation are the characteristic impedance, Z_c, and the propagation constant, γ. For example, as derived in section 3.2, for a distributed single conductor system (fig. 3.1), Z_c and γ are respectively given by:

$$Z_c = \sqrt{\frac{R + j\omega L}{G + j\omega C}} \tag{3.79}$$

$$\gamma = \sqrt{(R + j\omega L)(G + j\omega C)} \tag{3.80}$$

Both parameters are functions of frequency, even for constant distributed parameters R, L, G and C (except for lossless and distortionless lines). Fig. 3.4 shows the frequency dependence of the zero and positive sequence resistance and inductance of a three phase overhead line. It can be seen that the positive sequence inductance of the overhead lines is practically constant, while the positive sequence resistance remains more or less constant until skin effect in the conductors becomes noticeable. The zero sequence inductance and resistance are very much frequency dependent, due to skin effects in the earth return. As a result, both the characteristic impedance and the propagation constant are highly frequency dependent. Hence, in order to simulate an EHV transmission system correctly, a model with frequency dependent parameters must be used. There are various ways of handling the frequency dependence of the system parameters. In the following chapter, the frequency domain simulation approach and different time domain approaches based on travelling wave models will be described.

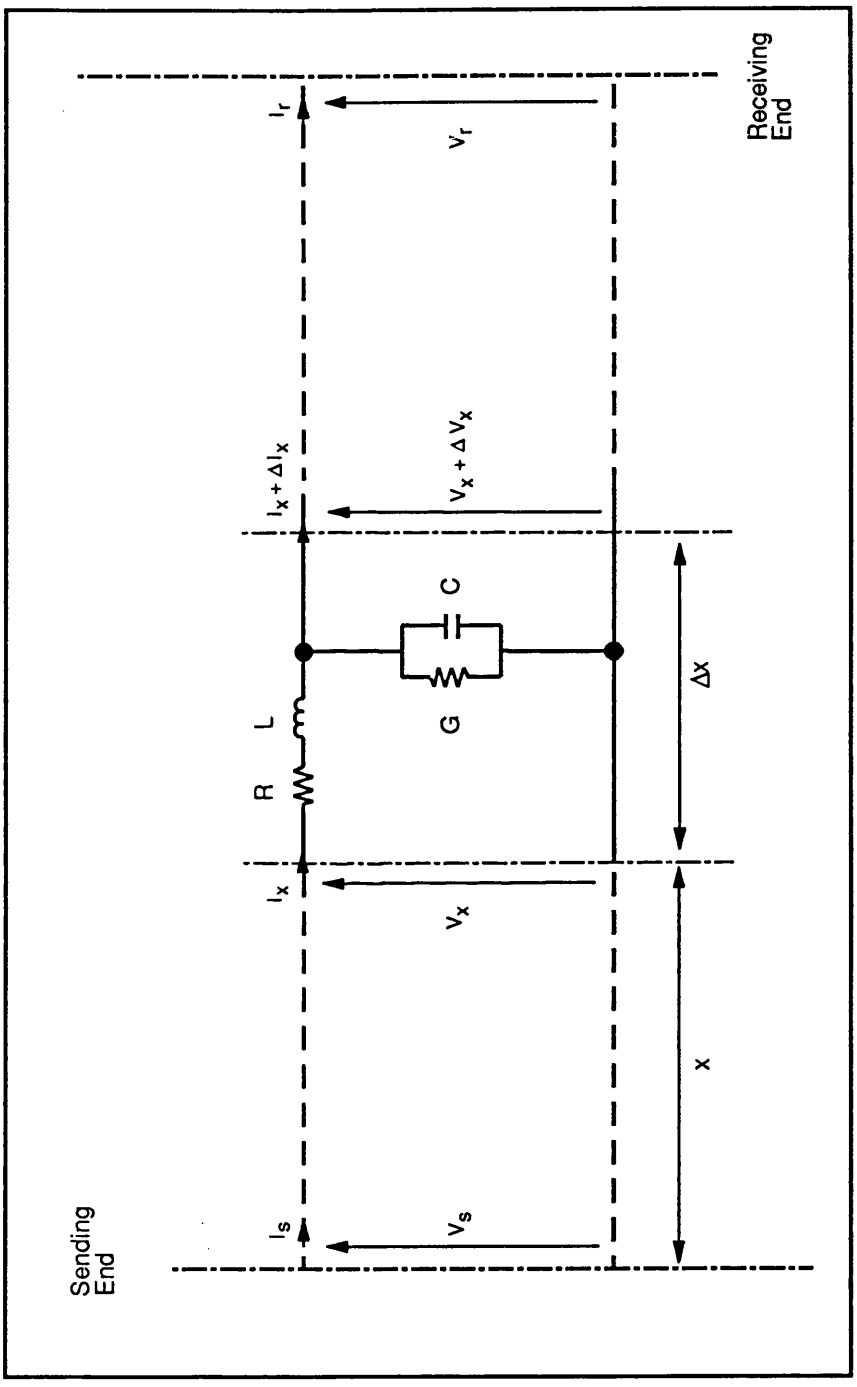

Figure 3.1: Distributed single conductor transmission line

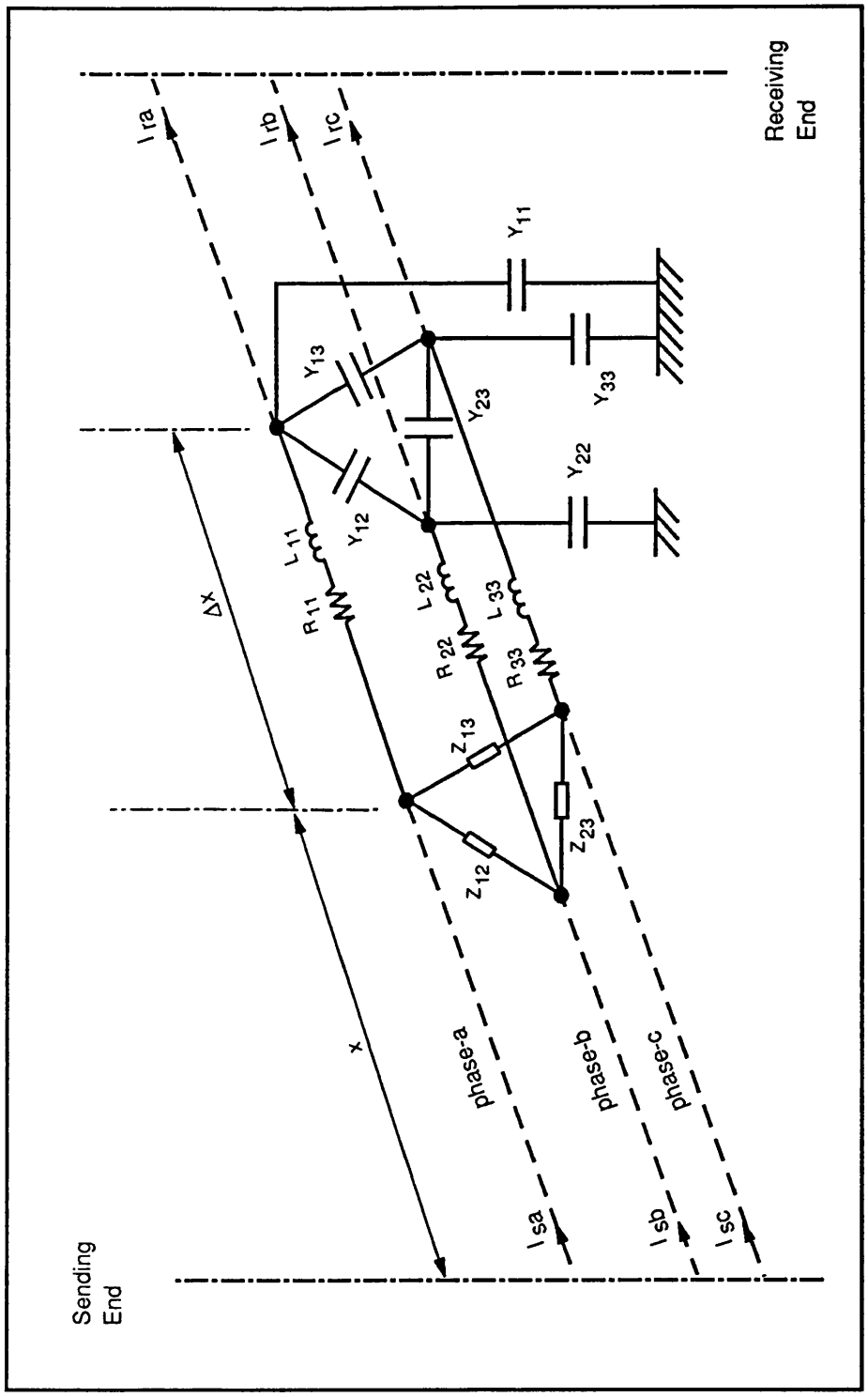

Figure 3.2: Distributed three phase transmission line

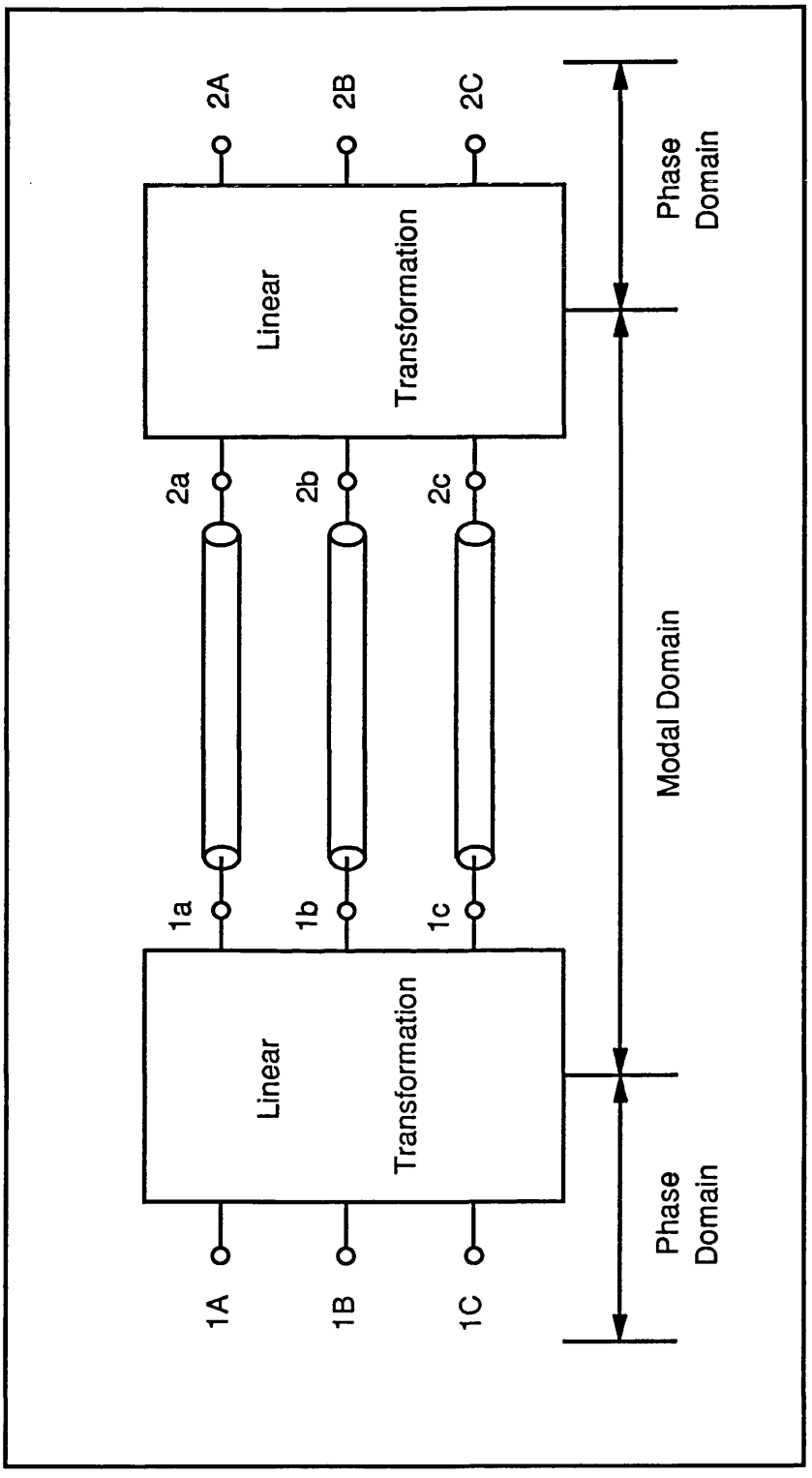

Figure 3.3: Modal decomposition of a three phase transmission system

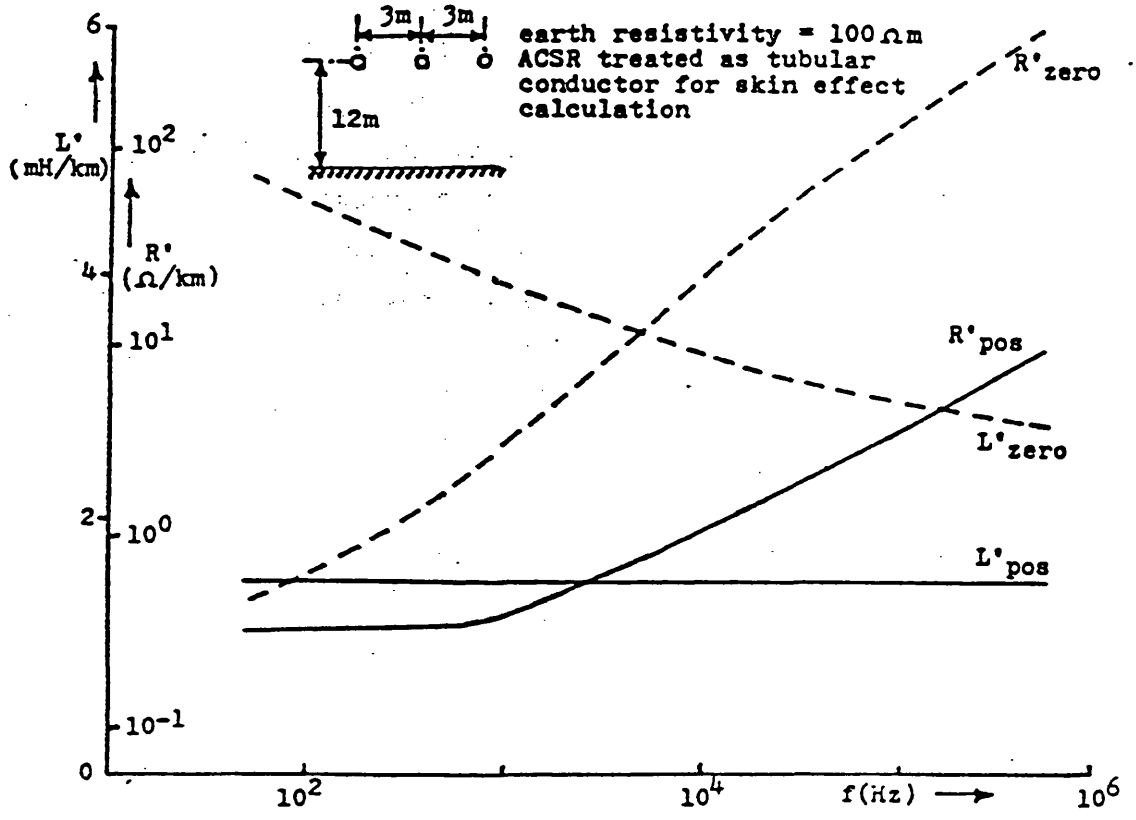

Figure 3.4: Frequency dependent parameters of a three phase line

Chapter 4

Electromagnetic transient models

4.1 Introduction

This chapter describes the techniques for deriving the electromagnetic transient models which are used in this project. Firstly, transient modelling using the frequency domain simulation approach [2] is described. And then, transient modelling using the EMTP program is described. In order to describe the transient models, three time domain approaches have to be considered sequentially. The first time domain approach is as described by Budner's admittance formulation [3]. The second one is Snelson's and Scott Meyer-Dommel's travelling functions approach [28, 29] and finally, the third one is Marti's approach [30]. It should be noted that these time domain approaches are mathematically quite similar. Snelson's model is based on Budner's formulation and overcomes some of the associated disapprovals. Marti further improves Snelson's model. In the project, Marti's model is used to produce frequency dependent line parameters for the electromagnetic transient simulation using EMTP. Finally, transient modelling

based on the time-convolutional technique is introduced. It should be emphasised that a simplified source model using constant voltage behind subtransient impedance is used for all the electromagnetic transient simulations.

4.2 Frequency domain formulation

4.2.1 Fundamental relationships overview

The theory of natural modes developed by Wedepohl [25] enables a solution to the system voltage steady-state equations given by eqn. 4.1 to be transformed into a series of independent differential equations of the form as shown in eqn. 4.2.

$$\frac{d^2V}{dx^2} = Z_{phase}Y_{phase}V \tag{4.1}$$

$$V = exp(-\psi x)V_i + exp(\psi x)V_r \tag{4.2}$$

where

ψ	=	$S\gamma S^{-1}$.
V_i	=	incident voltage transforms.
V_r	=	reflected voltage transforms.
Z_{phase}	=	series impedance matrix per unit length.
Y_{phase}	=	shunt admittance matrix per unit length.
γ	=	propagation constant matrix.

The polyphase surge impedance matrix is given by:

$$Z_c = S\gamma^{-1}S^{-1}Z_{phase} \tag{4.3}$$

and

$$\cosh(\psi x) = S\cosh(\gamma x)S^{-1} \tag{4.4}$$

4.2.2 Transfer matrix functions

Fig. 4.1 illustrates the basic arrangement for a faulted double end fed single conductor system. Fig. 4.2 illustrates the same case with a three phase transmission system. Such faulted transmission systems essentially consist of a network of cascaded sections and can be represented as shown in fig. 4.3. Two port transfer matrices are particularly useful in the solution of such problems. For example, the transfer matrix representing the line section up to the point of fault is given by eqn. 4.5, and this can be used in combination with the corresponding matrices representing the fault discontinuity and the line section between the fault and the receiving busbars, to yield a relationship between the currents and voltages at either end of the line. Eqn. 4.6 shows the multiplication process involved to yield the latter relationship.

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} E_{ff} \\ I_{fs} \end{bmatrix} \tag{4.5}$$

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} A_f & B_f \\ C_f & D_f \end{bmatrix} \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} V_r \\ I_r \end{bmatrix} \tag{4.6}$$

where

$$\begin{aligned} A_1 &= \cosh(\psi x) \\ B_1 &= \sinh(\psi x) Z_c \\ C_1 &= Y_c B_1 Y_c \\ D_1 &= Y_c A_1 Z_c \\ A_2 &= \cosh\left[\psi(L-x)\right] \\ B_2 &= \sinh\left[\psi(L-x)\right] Z_c \\ C_2 &= Y_c B_2 Y_c \\ D_2 &= Y_c A_2 Z_c \end{aligned}$$

The submatrices defining the transfer matrix representing the line section be-

yond the fault are found by substituting the length $(L - x)$ involved, and the matrix defining the fault discontinuity is formulated according to the type of fault simulated. Detailed derivations of the elements A, B, C & D are given in section 3.3.6.

4.2.3 Fault transient model

The basis of the method hinges upon representing the voltage at the fault point by the sum of two voltages V_{fs}, V_{ff} as shown in fig. 4.4. The first of these voltages, V_{fs}, is sinusoidal and is arranged to be equal to the steady-state voltage at the point of fault before disturbance. V_{ff} is a suddenly applied voltage, which, when added to V_{fs}, represents the postfault voltage. A solution may thus be obtained by performing two separate calculations in which the desired voltages and currents are evaluated when V_{fs} is applied to the energised system, and the superimposed voltage V_{ff} is applied to the line with all source voltages set at zero. The method is essentially one of superposition, and it should be noted that the steady-state voltage vector V_{fs} can be evaluated from a knowledge of the prefault voltages and currents at the terminating busbars. In fact, the vector V_{fs} is only needed insofar as it enables the value V_{ff} to be evaluated as a prerequisite to the second part of the computational process.

4.2.4 Formulation for 2-ended system

Prefault steady state

The vector V_{fs} is evaluated from a knowledge of the prefault voltages at the terminating busbars. Before the fault, the sending end and receiving end current and voltage are related by eqn. 4.7.

$$\begin{bmatrix} V_{ss} \\ I_{ss} \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} V_{rs} \\ I_{rs} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_{rs} \\ I_{rs} \end{bmatrix} \tag{4.7}$$

The sending end current and the fault point voltage before the fault are thus given by:

$$I_{ss} = \left[C - DB^{-1}A\right] V_{rs} + DB^{-1}V_{ss} \tag{4.8}$$

$$V_{fs} = \left[A_2 - B_2B^{-1}A\right] V_{rs} + B_2B^{-1}V_{ss} \tag{4.9}$$

Each of the above equations are evaluated at power frequency, because the prefault condition is essentially a steady-state one. Under this condition, the voltage and current vectors may be considered in phasor form and converted to the time domain without invoking the inverse Fourier transform.

Sound phase problem

To complete the solution, it is necessary to establish the response of the unenergised circuit to the superimposed voltage transform V_{ff}. A value is required for both faulted and sound phases. In the case of a faulted phase or phases, no difficulty exists. For example, considering a single-phase to earth fault which occurs on phase-a, the superimposed voltage, V_{ffa} is simply equal and opposite to the prefault voltage at the point of fault. For the unfaulted conductors, there is

no such direct means of knowing the appropriate value of superimposed voltage. However, the currents due to the application of the superimposed voltages are related by eqn. 4.10 [1]:

$$V_{ff} = -\left\{\left[\left(C_1 + D_1 Z_{ss}^{-1}\right)\left(A_1 + B_1 Z_{ss}^{-1}\right) + \left(C_2 + D_2 Z_{sr}^{-1}\right)\left(A_2 + B_2 Z_{sr}^{-1}\right)^1\right] + Z_f\right\} \left(I_{fsf} - I_{frf}\right) \tag{4.10}$$

The above equation involves 3×3 matrix for single circuit three phase systems. Thus, the above equation can be written in the following form:

$$\begin{bmatrix} V_{ffa} \\ V_{ffb} \\ V_{ffc} \end{bmatrix} = \begin{bmatrix} Z_{aa} & Z_{ab} & Z_{ac} \\ Z_{ba} & Z_{bb} & Z_{bc} \\ Z_{ca} & Z_{cb} & Z_{cc} \end{bmatrix} \begin{bmatrix} I_{fsfa} - I_{frfa} \\ I_{fsfb} - I_{frfb} \\ I_{fsfc} - I_{frfc} \end{bmatrix} \tag{4.11}$$

Consider now a single phase to earth fault involving phase-a, the currents I_{fsf} and I_{frf} will be equal in the b and c phases because they are healthy, and eqn. 4.11 takes the following simplified form:

$$\begin{bmatrix} V_{ffa} \\ V_{ffb} \\ V_{ffc} \end{bmatrix} = \begin{bmatrix} Z_{aa} \\ Z_{ba} \\ Z_{ca} \end{bmatrix} \left[I_{fsfa} - I_{frfa}\right] \tag{4.12}$$

With reference to eqn. 4.12, the voltage transform V_{ffa} is known, so that the superimposed voltages from each conductor to earth, E_{ff}, is obtained from eqn. 4.13.

$$E_{ff} = V_{ff} + Z_f \left(I_{fsf} - I_{frf}\right) \tag{4.13}$$

The final step in the process involves computing the spectrum of the voltage and/or current of interest. For example, the transformed superimposed voltages and currents at the sending end of the line are given by eqns. 4.14 and 4.15 [1] respectively.

$$V_{sf} = \left[B_1 D_1^{-1} Z_{ss}^{-1} + U\right]^{-1} \left[A_1 - B_1 D_1^{-1} C_1\right] E_{ff} \tag{4.14}$$

$$I_{sf} = -Z_{ss}^{-1} \left[B_1 D_1^{-1} Z_{ss}^{-1} + U\right]^{-1} \left[A_1 - B_1 D_1^{-1} C_1\right] E_{ff} \tag{4.15}$$

The above equations are converted to time variables via the inverse transform, and are added to the corresponding prefault steady-state variations to obtain

the complete fault-transient waveform. Similar techniques are employed in the analysis of other fault types. Interphase faults not involving earth are best dealt with by considering the superimposed voltages between phases. Apart from this, the analysis of such faults is very similar to that outlined in relation to earth faults.

Formulation of universal system matrices

During the process of fault inception and clearing, a transmission system is subjected to a wide range of frequency variations, and it is important to ensure that the response is evaluated over the whole frequency spectrum of importance. In order to take account of the frequency variation of the transmission system parameters, it is therefore necessary to implement the so-called *'superimposed'* simulation technique as developed by Johns and Aggarwal [2]. The standard superimposed circuit of the full system models (figs. 4.1 and 4.2) is represented in the frequency domain by the circuit of fig. 4.5. In order to effect a complete simulation, it is necessary to solve a total of $(n-1)$ such circuits. Let the suffix k, represents the kth superimposed circuit $(k = 2, \cdots, n)$. Then, the basic relationships of eqns. 4.16, 4.17 and 4.18 describe the circuit at each end of the line and at the fault point, respectively.

$$V_{sk} = -Z_{ss} I_{sk} - E_{sk} \tag{4.16}$$

$$V_{rk} = Z_{sr} I_{rk} + E_{rk} \tag{4.17}$$

$$V_{fk} = E_{fk} + Z_f \left[I_{fsk} - I_{frk} \right] = E_{fk} + Z_f I_{fk} \tag{4.18}$$

Furthermore, the line sections on each side of the fault are most conveniently represented in terms of the polyphase 2-port as defined in eqns. 4.19 and 4.20:

$$\begin{bmatrix} V_{sk} \\ I_{sk} \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} V_{fk} \\ I_{fsk} \end{bmatrix} \tag{4.19}$$

$$\begin{bmatrix} V_{fk} \\ I_{fsk} \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} V_{rk} \\ I_{rk} \end{bmatrix} \tag{4.20}$$

where A_1, B_1, C_1 and D_1, etc. are defined in section 4.2.2. The relationships in eqns. 4.16 to 4.20 effectively define a set of simultaneous equations relating transforms of the kth superimposed-circuit transform currents to the associated transform voltages across each hypothetical current generator and they can be arranged in the following alternative form:

$$\begin{bmatrix} I_{fk} \\ I_{sk} \\ I_{rk} \end{bmatrix} = \begin{bmatrix} Y_A & Y_B & Y_C \\ Y_D & Y_E & Y_F \\ Y_G & Y_H & Y_I \end{bmatrix} \begin{bmatrix} E_{ffk} \\ E_{sk} \\ E_{rk} \end{bmatrix} \tag{4.21}$$

$$\begin{bmatrix} E_{ffk} \\ E_{sk} \\ E_{rk} \end{bmatrix} = \begin{bmatrix} Z_A & Z_B & Z_C \\ Z_D & Z_E & Z_F \\ Z_G & Z_H & Z_I \end{bmatrix} \begin{bmatrix} I_{fk} \\ I_{sk} \\ I_{rk} \end{bmatrix} \tag{4.22}$$

Each of the submatrices in the foregoing two equations is defined in terms of the basic parameters (Z_f, A_1, Z_{ss}, etc.) of the system at any spectral frequency of interest. Notice that each element of the Y-matrix and Z-matrix of the preceding two equations is a 3×3 submatrix for single-circuit three phase configuration. Detailed definition of each element of the matrices is given in Appendix B. The 9×9 admittance matrix of eqn. 4.21 is essentially a universal relationship which can be computed and stored at all spectral frequencies of interest at the outset of a particular simulation study. The Z matrix in eqn. 4.22 is obtained by inverting the Y matrix, which is defined in eqn. 4.21.

Any fault sequence can be simulated based on the above formulation. Detailed numerical calculation techniques involved in simulating the fault sequence are given in Chapter 6.

4.3 EMTP modelling

Consider the basic double end fed transmission model as shown in fig. 4.2. With this particular model any type of earth fault can be treated by simulating the closure of the appropriate switches at the fault point. Fault break off or release is likewise simulated by opening the appropriate fault point switches in series with the total fault-path impedance (Z_{fa}, Z_{fb}, Z_{fc}). Pure interphase faults are likewise modelled by an appropriate interconnection of fault impedances and switches. Opening, and subsequent reclosure, of the circuit breakers is likewise simulated by the operation of the appropriate switches, according to the sequence associated with any type of autoreclosure cycle under consideration.

EMTP [6] is a time domain transient program, i.e., it solves the electromagnetic transient problem step by step in the time domain. The method, firstly, requires reduction of the whole system into its components, that is, combinations of equivalent node voltage source, branch current source, transmission line, resistance, inductance and capacitance. Secondly, nodal analysis of the network is performed and a set of equations which simulate the behaviour of each component is set up. Finally, the whole set of equations are solved numerically in the time domain and each new output is derived from a few already known past history terms.

Thus, referring to the basic system arrangement as shown in fig. 4.2, the sending end source and receiving end source are respectively simulated using an equivalent constant voltage source behind subtransient impedance. The circuit breakers and the fault path switches are easily simulated using time-varying resistances, e.g., $R = 0$ when closed and $R = \infty$ when open. The only and the most complicated component left is the simulation of equivalent transmission line.

It has long been recognized that one of the most important aspects in the modelling of transmission lines for electromagnetic transient studies is to account for the frequency dependence of line parameters and for the distributed nature of losses. The following section describes detailed frequency dependent line modelling in the time domain. Detailed numerical transient calculations of the overall system using the EMTP is elaborated later in Chapter 6.

4.4 Time domain formulations of frequency dependent transmission lines

The problem related to the inclusion in the time domain of both the frequency variation of the parameters and the distributed nature of the losses is twofold. On the one hand, the problem can be solved easily in the frequency domain. But, on the other hand, a general transient solution for a complete system is much more convenient in the time domain. Which particular solution technique is used is influenced by numerous important factors, such as computer time, numerical stability, accuracy, etc.

Present day understanding and formulation of the problem can be reviewed through the main approaches which are sometimes mathematically quite similar, although important differences with regard to the computation techniques exist. The train of thought associated with the equivalent circuit formulation can be traced through the works by A. Budner [3], J.K. Snelson [28], W.S. Meyer & H.W. Dommel [29] and J. Marti [30].

4.4.1 Budner's admittance formulation

Treating a basic two-ended single conductor system as a two-port network (fig. 4.6a), A. Budner obtains the equations of the network in the frequency domain as follows:

$$I_k = Y_{kk}V_k + Y_{km}V_m \tag{4.23}$$

$$I_m = Y_{mk}V_k + Y_{mm}V_m \tag{4.24}$$

where subscripts k and m denote node-'k' and node-'m' respectively as illustrated in fig. 4.6.

Comparing these equations with the corresponding frequency domain solution, the following relations can be obtained:

$$Y_{kk} = Y_{mm} = Z_c^{-1}\coth(\gamma l) \tag{4.25}$$

$$Y_{km} = Y_{mk} = -Z_c^{-1}\mathrm{cosech}(\gamma l) \tag{4.26}$$

where l is the distance between node k and m.

The inverse Fourier transforms of equations 4.23 and 4.24 yield:

$$i_k(t) = y_{kk}(t) * v_k(t) + y_{km}(t) * v_m(t) \tag{4.27}$$

$$i_m(t) = y_{km}(t) * v_k(t) + y_{kk}(t) * v_m(t) \tag{4.28}$$

After evaluating the convolution integrals for solving equations 4.27 and 4.28, $i_k(t)$ and $i_m(t)$ can be expressed as functions of their corresponding voltages and of values known from previous steps:

$$i_k(t) = y_s v_k(t) + i_{k,hist}(t-\tau) \tag{4.29}$$

$$i_m(t) = y_s v_m(t) + i_{m,hist}(t-\tau) \tag{4.30}$$

In the preceding equations, y_s is a constant, and $i_{k,hist}$ and $i_{m,hist}$ are functions of past values.

To calculate the time domain form of the admittances, Budner uses a fast Fourier transform, but this is faced with numerous difficulties. The discrete inverse Fourier transform, imposes a bandwidth limitation on the frequency function. In addition, a smaller $\Delta\omega$ and consequently a longer computation time is needed.

Functions $y_{kk}(t)$ and $y_{km}(t)$ are made up of a series of spikes. They are at first very sharp and then become very flat. This feature of the admittances very much complicates the convolution process which has to be performed at each time step of the complete network solution and tends to become inaccurate and very time consuming.

The form of the y-weighting-functions can be visualized from the system conditions as illustrated in fig. 4.6b. The voltage impulses applied at node k travels toward node m. The time impulse has a uniform frequency distribution and different frequencies have different travelling times and attenuations. Thus, by the time, the impulse reaches node m, it has become a spike. This is the first peak in y_{km}. After reflection, the wave travels back to node k and reaches it after corresponding travelling times and attenuations, making up the first spike in the function y_{kk}. The process of reflections and wave travelling forward and backward continues for a very long time.

4.4.2 Snelson's and Scott Meyer-Dommel's travelling functions approach

Looking for simpler weighting functions than those in Budner's approach [3], J.K. Snelson [28] defines a new set of variables to relate currents and voltages at the ends of the line. Snelson's idea has been further developed by W. Scott Meyer and H.W. Dommel and implemented in the EMTP [6].

The new variables defined by Snelson are as follows:

- For forward travelling functions:

$$f_k(t) = v_k(t) + R_1 i_k(t) \tag{4.31}$$

$$f_m(t) = v_m(t) + R_1 i_m(t) \tag{4.32}$$

- For backward travelling functions:

$$b_k(t) = v_k(t) - R_1 i_k(t) \tag{4.33}$$

$$b_m(t) = v_m(t) - R_1 i_m(t) \tag{4.34}$$

Snelson defines R_1 as $\lim_{\omega \to \infty} Z_c(\omega)$, where $Z_c(\omega)$ is the line characteristic impedance. Relations 4.31 and 4.32 can be visualized physically from fig. 4.7.

The functions b_k and b_m contain the rest of the information determining the internal behaviour of the line. The time domain functions f and b can be transformed into the frequency domain:

- For forward travelling functions:

$$F_k(\omega) = V_k(\omega) + R_1 I_k(\omega) \tag{4.35}$$

$$F_m(\omega) = V_m(\omega) + R_1 I_m(\omega) \tag{4.36}$$

- For backward travelling functions:

$$B_k(\omega) = V_k(\omega) - R_1 I_k(\omega) \tag{4.37}$$

$$B_m(\omega) = V_m(\omega) - R_1 I_m(\omega) \tag{4.38}$$

Comparing the last four equations with the exact line solution in the frequency domain, the following expressions can be written:

$$B_k(\omega) = A_1(\omega)F_m(\omega) + A_2(\omega)F_k(\omega) \tag{4.39}$$

$$B_m(\omega) = A_1(\omega)F_k(\omega) + A_2(\omega)F_m(\omega) \tag{4.40}$$

where

$$A_1(\omega) = \frac{1}{\cosh(\gamma l) + \frac{1}{2}\left[\frac{Z_c}{R_1} + \frac{R_1}{Z_c}\right]\sinh(\gamma l)} \tag{4.41}$$

$$A_2(\omega) = \frac{1}{2}\left[\frac{Z_c}{R_1} - \frac{R_1}{Z_c}\right]\sinh(\gamma l)A_1(\omega) \tag{4.42}$$

The frequency dependence of the characteristic impedance $Z_c(\omega)$ and the propagation constant $\gamma(\omega)$ in equations 4.39 and 4.40 are included in the functions A_1 and A_2. Transforming these equation into the time domain, the following equations can thus be obtained:

$$b_k(t) = a_1(t) * f_m(t) + a_2(t) * f_k(t) \tag{4.43}$$

$$b_m(t) = a_1(t) * f_k(t) + a_2(t) * f_m(t) \tag{4.44}$$

where $a_1(t)$ and $a_2(t)$ are the weighting functions for the time domain convolutions.

After evaluating the convolutions in equations 4.43 and 4.44, equations 4.33 and 4.34 give, at each time step, the following equivalent line representation

for node k and m:

$$i_k(t) = \frac{1}{R_1} v_k(t) + i_{k,hist}(t - \Delta t) \tag{4.45}$$

$$i_m(t) = \frac{1}{R_1} v_m(t) + i_{m,hist}(t - \Delta t) \tag{4.46}$$

where $i_{k,hist}$ and $i_{m,hist}$ are known from the past history values.

The physical interpretation of the weighting functions $a_1(t)$ and $a_2(t)$ can be visualized from fig. 4.8. As can be seen, these weighting functions are generated by the voltage impulse $\delta(t)$ travelling along the line. In comparison with Budner's approach, the reflections are much smaller and the spikes in the weighting functions get attenuated much faster. Since the functions $a_1(t)$ and $a_2(t)$ decay much faster than the functions y_{kk}, y_{mm} and y_{km}, the corresponding time domain convolutions at each step of the network solution require much less computation time. Also, the frequency domain form of these functions is less oscillatory and thus, consequently, facilitates the numerical evaluation of the corresponding time domain forms.

4.4.3 Marti's approximation by rational functions

The previous method involves difficulties in the evaluation of the weighting functions $a_1(t)$ and $a_2(t)$. Since the resistance, especially for the zero sequence mode, increases with frequency, the high frequency components are attenuated much faster than the low frequency components. This means that the low frequency components will have a large effect on the tail portion. The amplitude of the reflections of the applied voltage is also larger at lower frequencies since the resistance R_1 is much closer to the characteristic impedance at the higher frequencies. This is an obvious result of the definition of R_1 as being equal to $\lim_{\omega \to \infty} Z_c$.

Reasoning on the aforementioned mentioned problem, Marti [30] decided to replace the simple resistance R_1 at the ends of the line by a network whose frequency response matches the line characteristic $Z_c(\omega)$. In this case, the voltage impulse applied at node k would not be reflected back from node m. Hence, the weighting function $a_1(t)$ would have only one spike and the weighting function $a_2(t)$ would be equal to zero. The replacement of the resistance R_1 by the network Z_{eq}, which approximates the characteristic impedance $Z_c(\omega)$, is schematically shown in fig. 4.9.

After replacing R_1 with Z_{eq}, equations 4.35 to 4.38 have the following forms:

- For forward travelling functions:

$$f_k(t) = v_k(t) + e_k(t) \tag{4.47}$$

$$f_m(t) = v_m(t) + e_m(t) \tag{4.48}$$

- For backward travelling functions:

$$b_k(t) = v_k(t) - e_k(t) \tag{4.49}$$

$$b_m(t) = v_m(t) - e_m(t) \tag{4.50}$$

where $e_k(t)$ and $e_m(t)$ are voltages across equivalent network Z_{eq} due to the currents $i_k(t)$ and $i_m(t)$, respectively. If the impedance Z_{eq} which approximates the characteristic impedance $Z_c(\omega)$ is assumed to be linear, the preceding four equations can be transformed into frequency domain as follows:

- For forward travelling functions:

$$F_k(\omega) = V_k(\omega) + Z_{eq} I_k(\omega) \tag{4.51}$$

$$F_m(\omega) = V_m(\omega) + Z_{eq} I_m(\omega) \tag{4.52}$$

- For backward travelling functions:

$$B_k(\omega) = V_k(\omega) - Z_{eq} I_k(\omega) \tag{4.53}$$

$$B_m(\omega) = V_m(\omega) - Z_{eq} I_m(\omega) \tag{4.54}$$

Comparing the four preceding equations with the general line solution in the frequency domain, the following equations can be written:

$$B_k(\omega) = A_1(\omega) F_m(\omega) \tag{4.55}$$

$$B_m(\omega) = A_1(\omega) F_k(\omega) \tag{4.56}$$

where

$$A_1(\omega) = e^{-\gamma(\omega) l} = \frac{1}{\cosh[\gamma(\omega) l] + \sinh[\gamma(\omega) l]} \tag{4.57}$$

The time domain form of equations 4.55 and 4.56 is given by the convolution integrals:

$$b_k(t) = \int_{\tau}^{\infty} f_m(t-u) a_1(u) du \tag{4.58}$$

$$b_m(t) = \int_{\tau}^{\infty} f_k(t-u) a_1(u) du \tag{4.59}$$

where τ represents the travelling time of the fastest frequency component of the applied impulse. In a discrete numerical solution, if the time step Δt is smaller than τ, the values of $f_k(t)$ and $f_m(t)$ in equations 4.58 and 4.59 are known at discrete points from the previous time steps and the integrals 4.58 and 4.59 can be evaluated. In this Marti approach, the problem of the tail effects is eliminated, the numerical integration in the evaluation of the history functions is much more efficient and the weighting function $a_2(t)$ is eliminated.

Here the key point of this approach is reached with the most important consequences in the contemporary situation of the frequency dependent modelling. The approach, however, encounters three problems:

- the synthesis of the equivalent network which approximates the characteristic impedance $Z_c(\omega)$,
- the evaluation of the weighting function and
- the synthesis of the transformation matrices S and Q.

Marti simulates the characteristic impedance $Z_c(\omega)$ using a Foster I network representation. For the synthesis of Z_{eq}, the tabular function $Z_c(\omega)$, which takes into account the frequency dependent parameters $Z(\omega)$ and $Y(\omega)$ via Carson's, Wise's or some other formulae [19], is approximated in the complex plane by the rational function of the following form:

$$Z_{eq} = \frac{N(s)}{D(s)} = \frac{(s+z_1)(s+z_2)\cdots(s+z_n)}{(s+p_1)(s+p_2)\cdots(s+p_n)} \tag{4.60}$$

where the break points z_i and p_i of this function are real, positive and simple. The parameters of Foster I representation are obtained by expanding the preceding equation into a series of simple fractions.

To obtain current sources, the expressions given by equations 4.58 and 4.59 must be evaluated. The process of evaluating convolution integrals from equations 4.58 and 4.59 can be accelerated if the corresponding weighting function can be expressed as a sum of exponential terms, that is, if the convolution integral, in general, has the following form at time step t:

$$s(t) = \int_{\tau}^{\infty} f(t-u)ke^{-\alpha(u-\tau)}du \tag{4.61}$$

Then $s(t)$ can be directly obtained from the known values $s(t-\Delta t)$ at the previous time step and the known history of f at τ and $(\tau+\Delta t)$.

Since the weighting function $a_1(t)$ is identical to zero for $t < \tau$, the propagation

response in the time domain equals:

$$a_1(t) = p(t - \tau) \tag{4.62}$$

The last expression means only that the function $a_1(t)$ is displaced toward the origin for τ time units. The corresponding frequency form of the last equation is:

$$A_1(\omega) = P(\omega)e^{-j\omega\tau} \tag{4.63}$$

The function $P(s)$ corresponding to $P(\omega)$ in the complex plane is approximated by a rational function of the form:

$$P(s) = \frac{N(s)}{D(s)} = H\frac{(s+z_1)(s+z_2)\cdots(s+z_n)}{(s+p_1)(s+p_2)\cdots(s+p_n)} \tag{4.64}$$

Since $A_1(\omega)$ corresponds to the response of a passive physical system and tends to zero when w tends to infinity, the number of zeros in formula 4.64 must be smaller than the number of poles, and the real part of the poles must lie in the left side of the complex plane. After partitioning the expression 4.64 into a series of simple functions, it is easy to obtain the time domain form of the weighting function $a_1(t)$.

4.5 Time convolutional modelling

Following early work by Budner [3] and by Umoto and Hara [18], the common theme was that of expressing nonlinear dependencies in the frequency-domain in impulse responses in the time-domain. Essentially, there are two stages of analysis. The first is a preparatory phase in which impulse response functions are formed. This arises only once for any given transmission system, following which the impulse responses found for it are held with other relevant transmission line data. The second stage is then that in which electromagnetic transient analysis is carried out and this requires evaluations of time convolutional integrals.

4.5.1 Fault transient modelling

Following the same analysis as given in section 4.2 for the frequency domain approach, the fault transient model is firstly established in the frequency domain. The system equations are given in equations 4.21 and 4.22 in the frequency domain. Expressing these equations in the corresponding time domain, the following set of equations are obtained:

$$\begin{bmatrix} i_f(t) \\ i_s(t) \\ i_r(t) \end{bmatrix} = \begin{bmatrix} y_a & y_b & y_c \\ y_d & y_e & y_f \\ y_g & y_h & y_i \end{bmatrix} * \begin{bmatrix} e_{ff}(t) \\ e_s(t) \\ e_r(t) \end{bmatrix} \tag{4.65}$$

$$\begin{bmatrix} e_{ff}(t) \\ e_s(t) \\ e_r(t) \end{bmatrix} = \begin{bmatrix} z_a & z_b & z_c \\ z_d & z_e & z_f \\ z_g & z_h & z_i \end{bmatrix} * \begin{bmatrix} i_f(t) \\ i_s(t) \\ i_r(t) \end{bmatrix} \tag{4.66}$$

where the symbol $*$ stands for convolution. Each of the element of the system matrices $[y]$ and $[z]$ in equations 4.65 and 4.66 represents the corresponding impulse response function of the system matrices $[Y]$ and $[Z]$ in the equations 4.21 and 4.22.

The following example is used to illustrate the use of the above two matrices equations. Suppose that $e_{ff}(t)$, $e_s(t)$ and $e_r(t)$ are known time domain functions, the fault transient of $i_f(t)$, $i_s(t)$ and $i_r(t)$ can be calculated directly in the time domain as follows:

$$i_f(t) = y_a * e_{ff}(t) + y_b * e_s(t) + y_c * e_r(t) \tag{4.67}$$

$$i_s(t) = y_d * e_{ff}(t) + y_e * e_s(t) + y_f * e_r(t) \tag{4.68}$$

$$i_r(t) = y_g * e_{ff}(t) + y_h * e_s(t) + y_i * e_r(t) \tag{4.69}$$

Detailed numerical calculation techniques involving fault sequence simulation using time convolutional technique are given in Chapter 6.

4.6 Simplified generator source model

The simplified power system source is modelled as a constant voltage connected in series with an equivalent subtransient impedance, Z_s. Z_s can be expressed as a resistance (R_s) connected in series with an inductance (L_s), i.e.,

$$Z_s = R_s + j\omega L_s \tag{4.70}$$

where $\omega = 2\pi f$ and f is the power frequency.

The Q-factor is defined as:

$$Q = \frac{j\omega L_s}{R_s} \tag{4.71}$$

R_s is calculated as:

$$R_s = \frac{V_{rms}^2}{S\sqrt{1+Q^2}} \tag{4.72}$$

where V_{rms} is the rated line voltage and S is the equivalent source capacity.

L_s can be calculated as:

$$L_s = \frac{QR_s}{\omega} \tag{4.73}$$

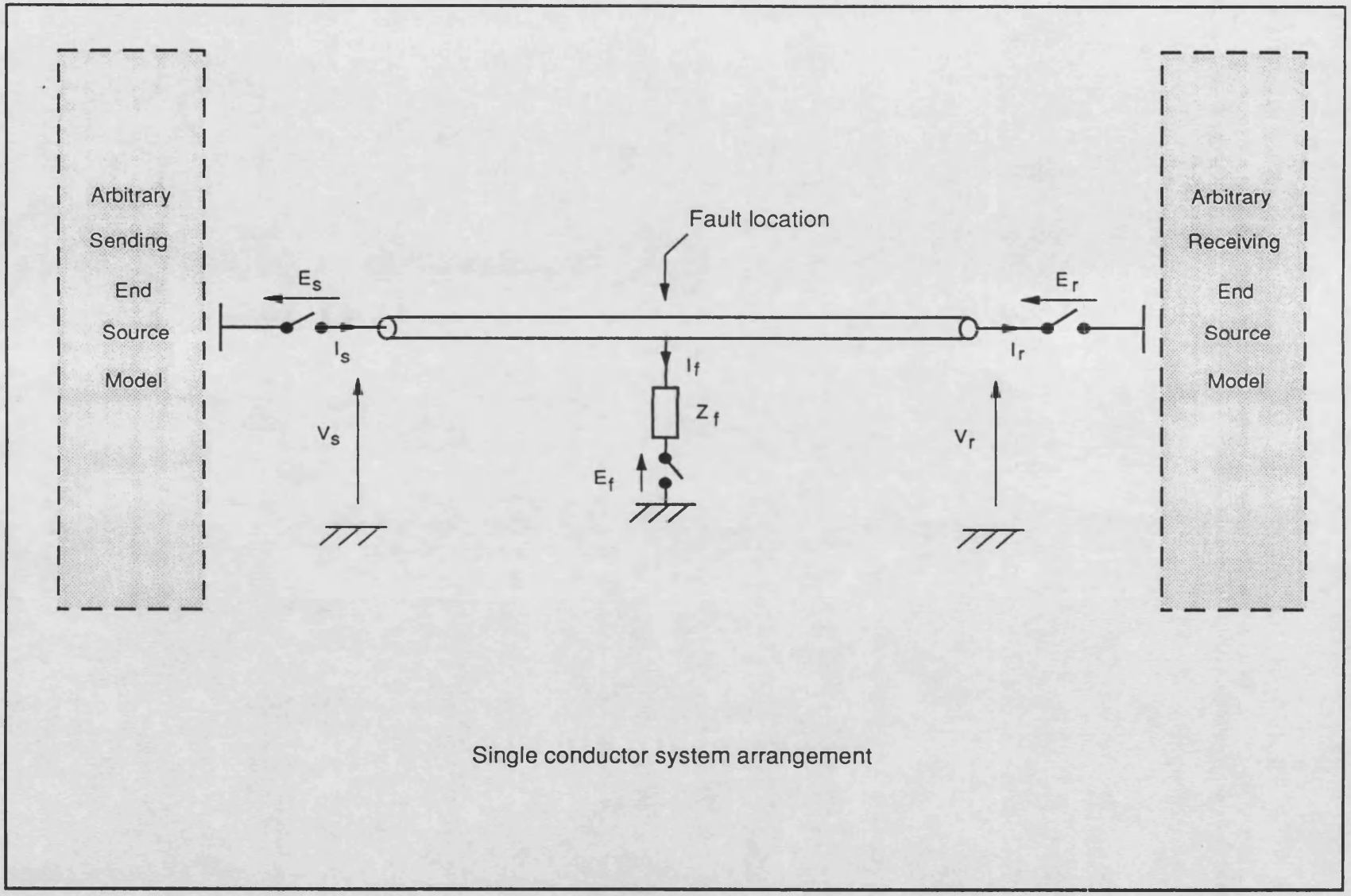

Figure 4.1: Single conductor system arrangement

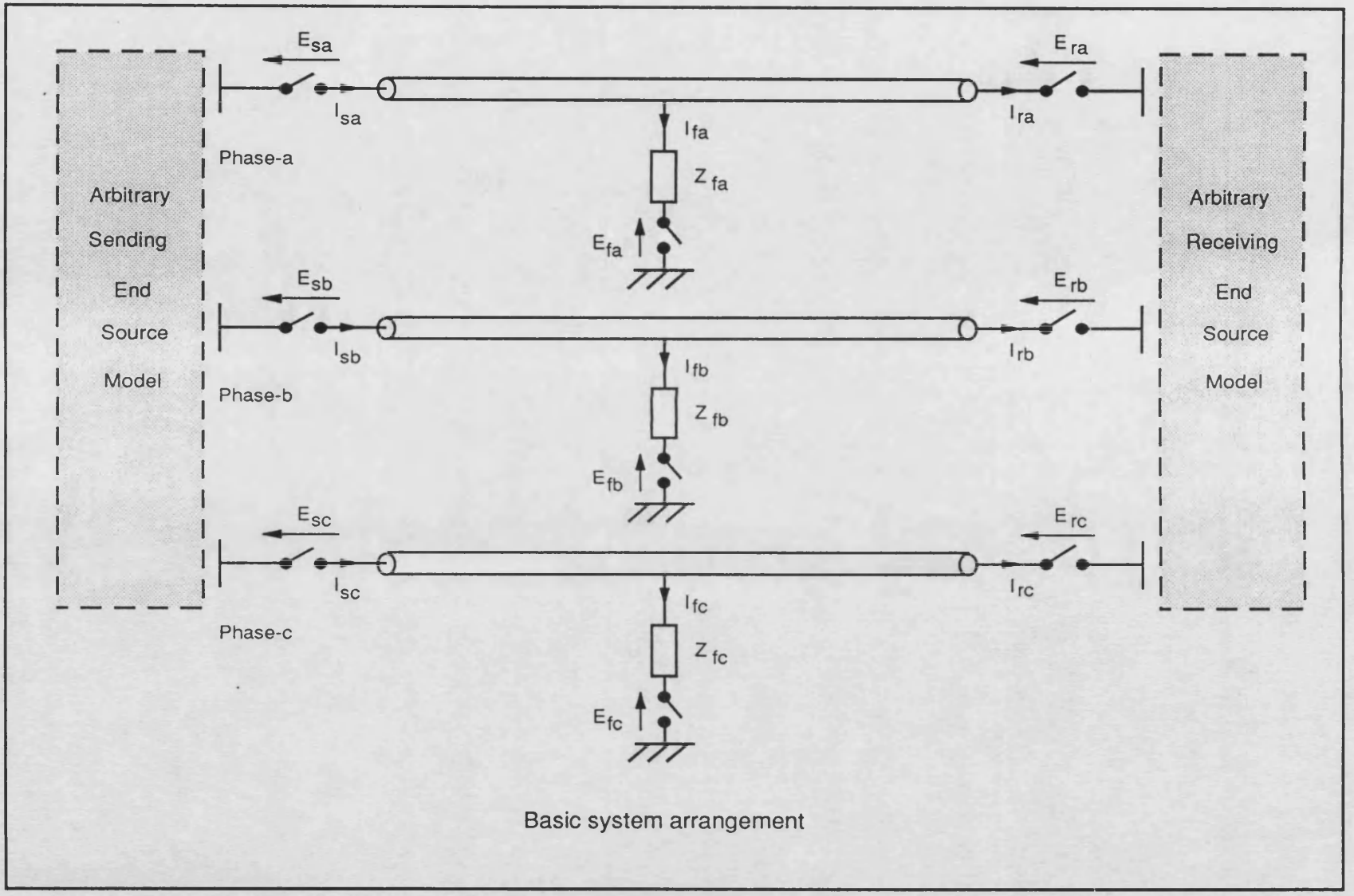

Figure 4.2: Three phase system arrangement

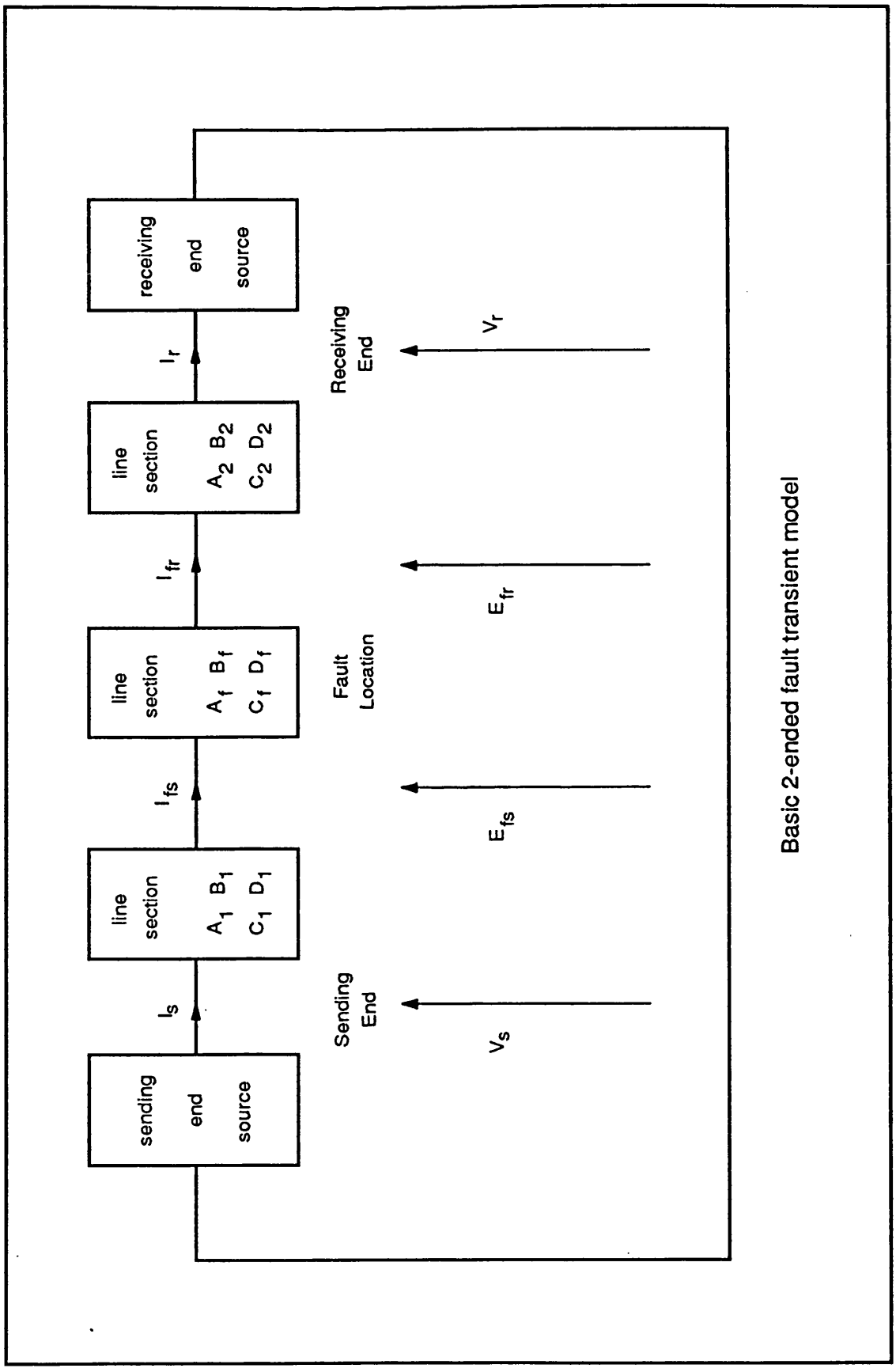

Figure 4.3: Basic 2-ended fault transient model

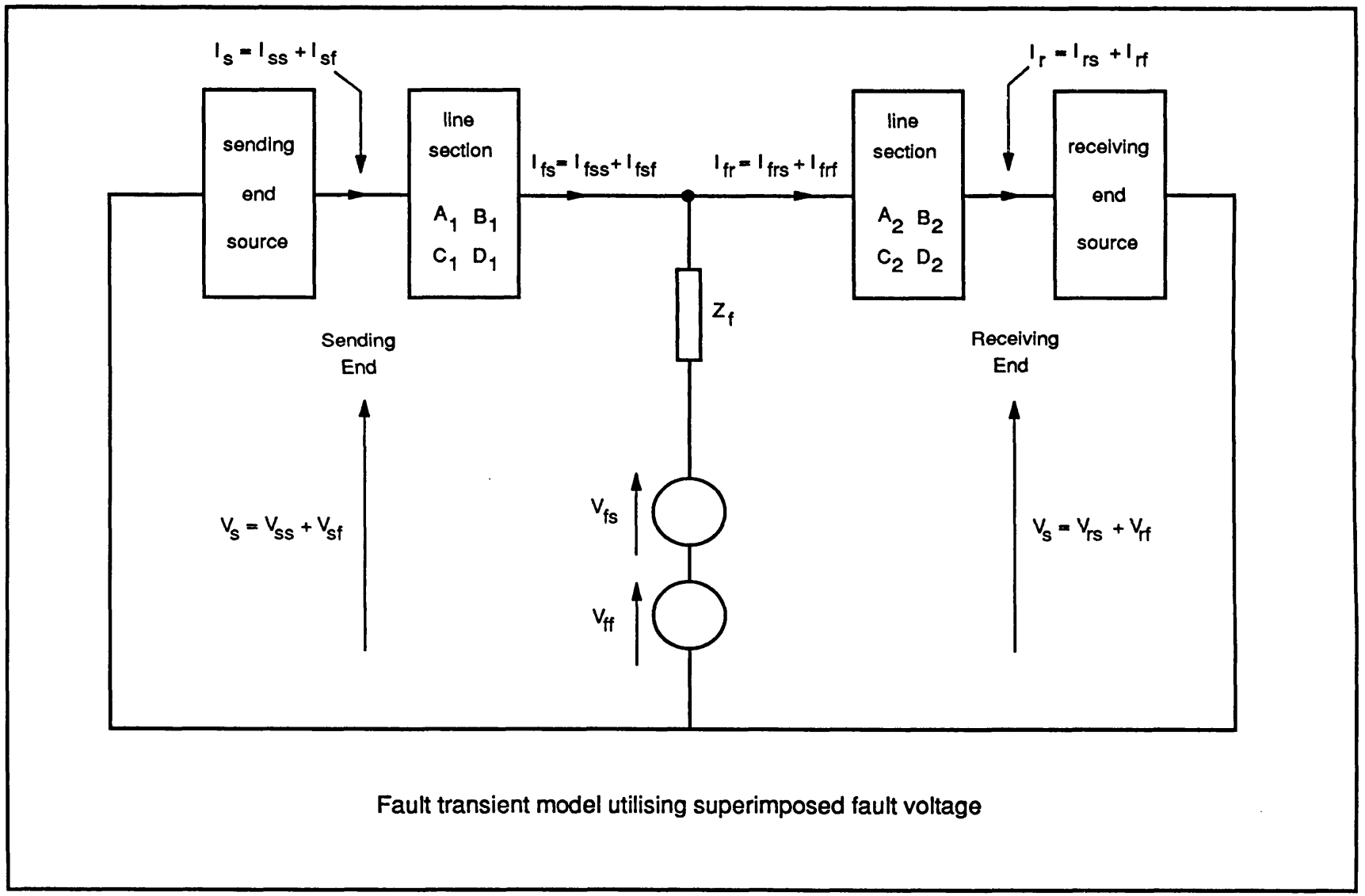

Figure 4.4: Fault transient model utilising superimposed fault voltage

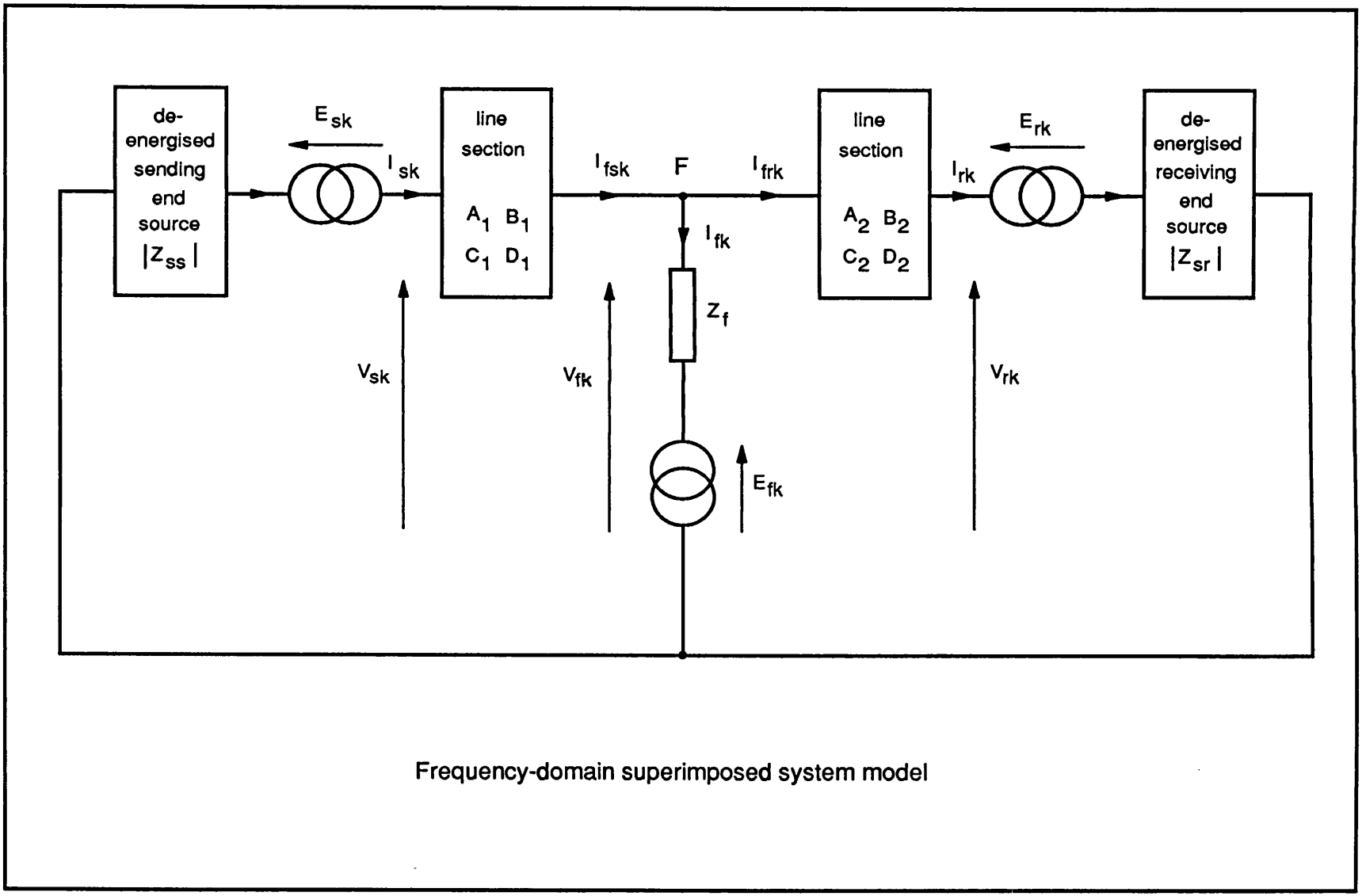

Figure 4.5: Frequency domain superimposed system model

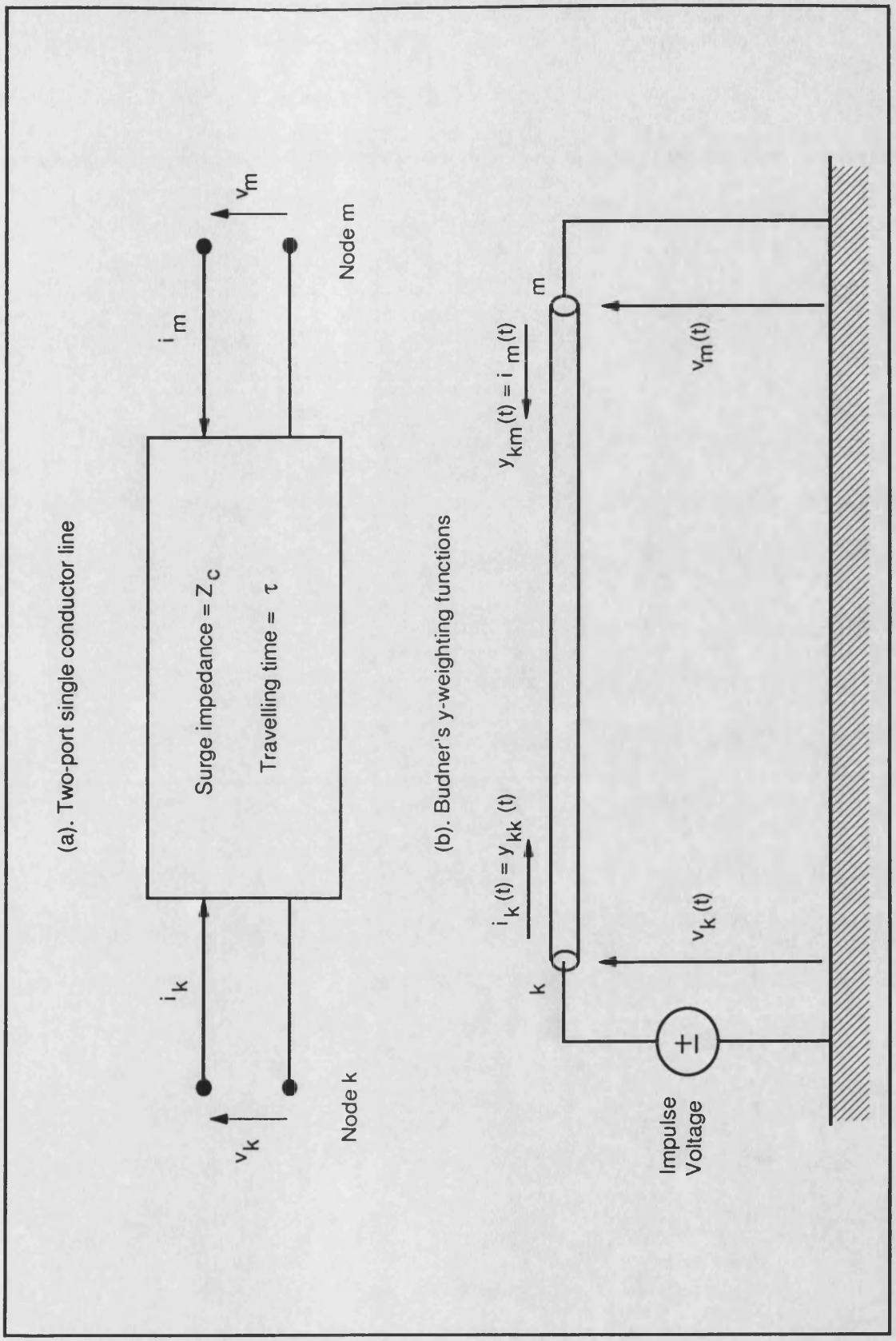

Figure 4.6: (a). Two-port single conductor line. (b). Visualization of Budner's y-weighting functions.

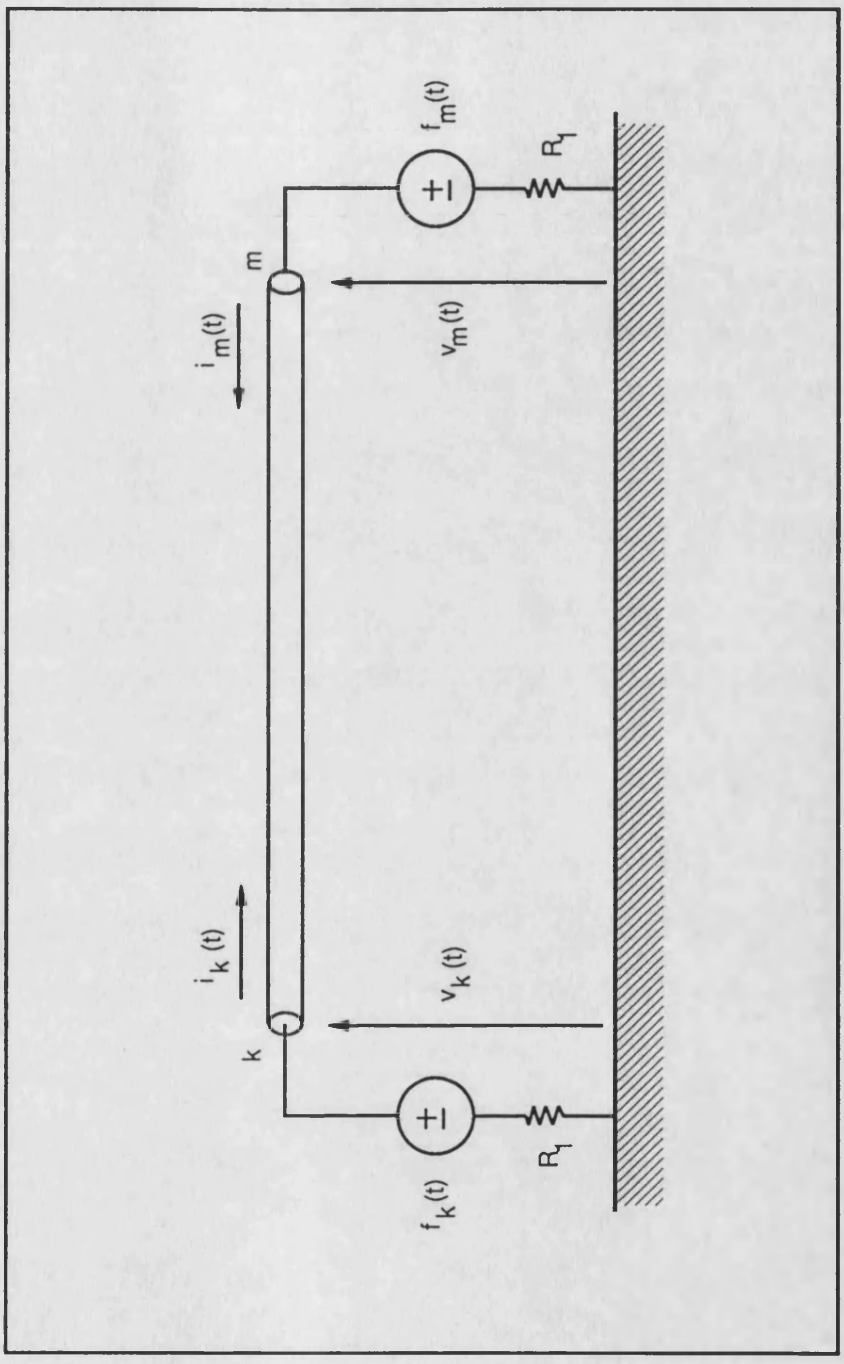

Figure 4.7: Visualization of Snelson's travelling wave functions

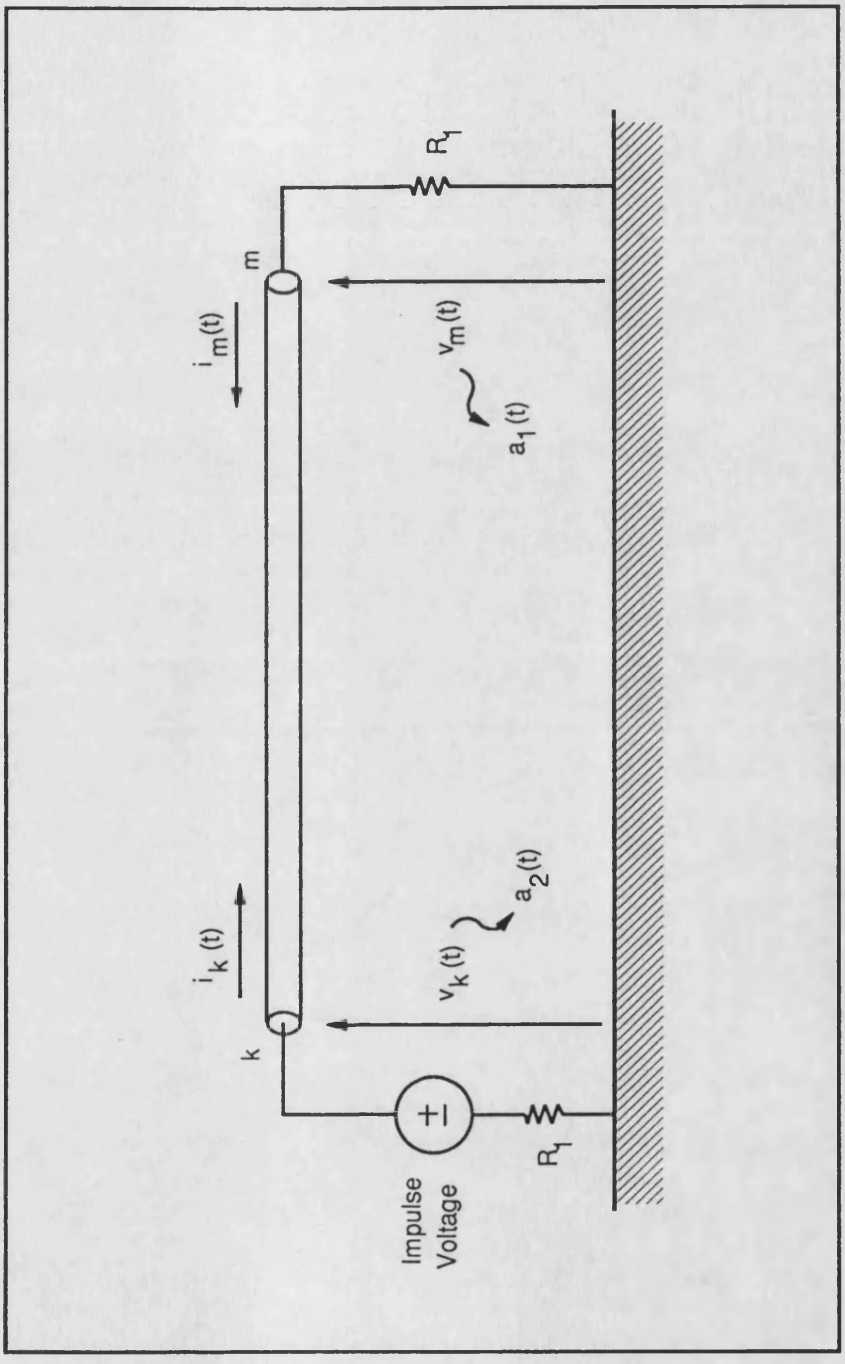

Figure 4.8: Visualization of Snelson's $a_1(t)$ and $a_2(t)$ weighting functions

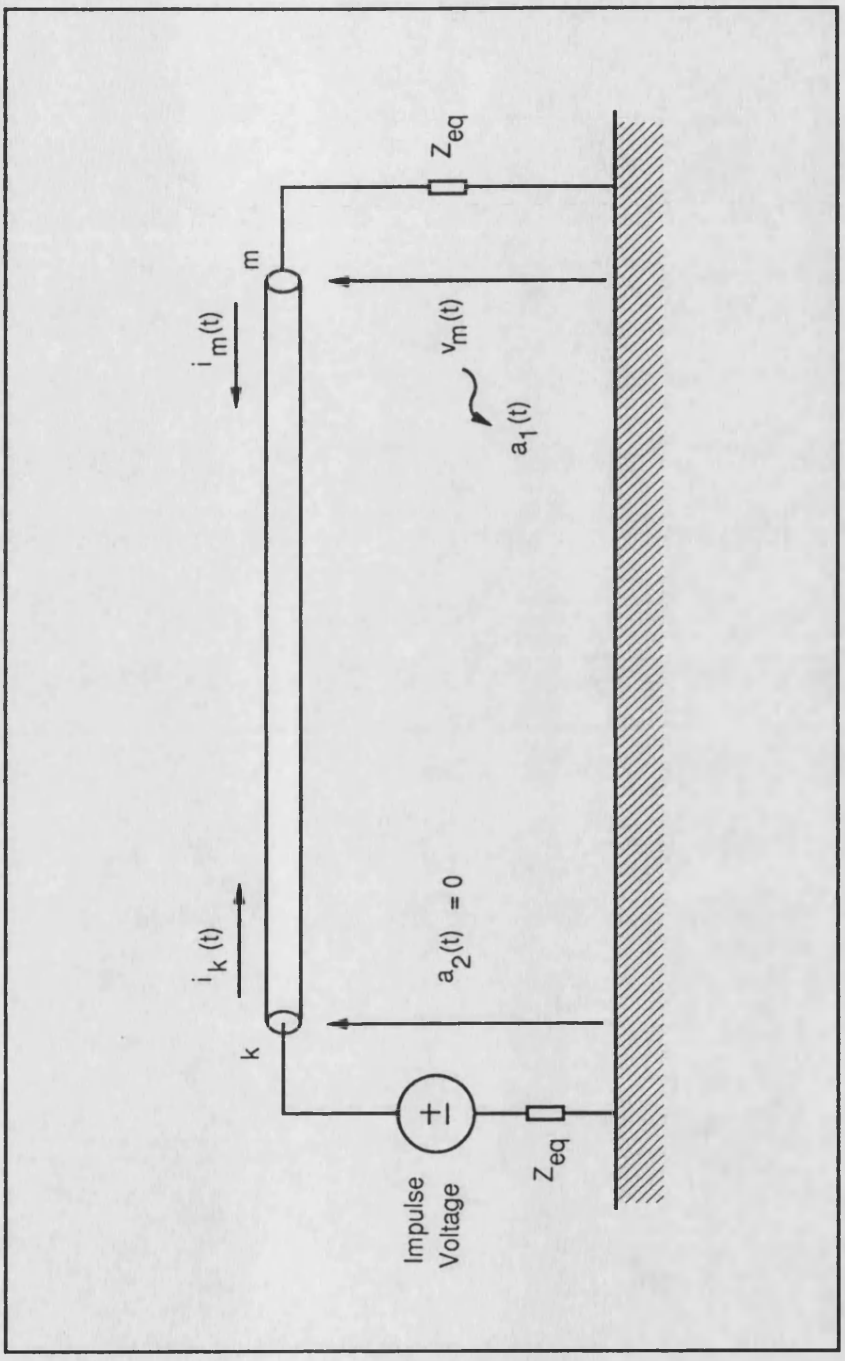

Figure 4.9: J. Marti's modifications to Snelson's model

Chapter 5

Electromechanical transient models

5.1 Introduction

A simplified source model, such as using a constant voltage behind subtransient impedance, can give acceptable accuracy for the fault transients during the prefault clearance period where the high frequency components are dominant. However, during the postfault clearance period, i.e., after the fault has been cleared, the low frequency components are dominant. Simplified source model may thus produce erroneous results. Furthermore, coupled with the fast development of very high speed microprocessor based protection schemes, modern power systems are able to operate closer to their stability limits. This in turn requires more detailed representation of the relevant dynamic properties of the power system elements so that accurate information can be used to validate and further develop better protection schemes. A detailed power system source model, which consists of an actual generator model, the AVR, the governor control system and the generator transformer, is thus required for achieving a better degree of ac-

curacy, in particular, to be able to produce the low frequency effects due to, for example, the pole-slippings of the generator.

This chapter describes the mathematical formulations of each element of a typical single machine power system source model as illustrated in fig. 5.1.

5.2 Synchronous generator

A synchronous generator model with three phase armature windings on the stator and a dc field winding with one or more pole pairs on the rotor is used. In general, it consists of electrical and mechanical representations [6, 31, 32].

5.2.1 Electrical representation

Fig. 5.2 shows the electrical part of the model. It has been assumed that the generator consists of the following magnetically coupled windings:

1. Three armature windings *a, b, c*, one per phase, connected to the network. The space displacement among the a-b-c phases is 120 electrical degrees.
2. One field winding f which produces flux in the direct axis.
3. One hypothetical winding kd on the direct axis which represents the damper bar effects.
4. One hypothetical winding g on the quadrature axis representing the effects produced by eddy-currents.

5. One hypothetical winding *kq* on the quadrature axis representing damper bar effects.

The dynamic synchronous machine is internally balanced with respect to the armature phases. The armature windings are assumed to be Wye-connected, with a possible neutral to ground impedance. To make the electrical equations manageable, a number of idealized characteristics are assumed, which are reasonable for system studies. These assumptions for an 'ideal synchronous machine' [33] are:

- The resistance of each winding is constant.
- The permeance of each portion of the magnetic circuit is constant.
- The armature windings are symmetrical with respect to each other.
- The electric and magnetic circuits of the field structure are symmetrical about the direct or quadrature axis.
- The self inductance of each winding on the field structure *(f, g, kd, kq)* is constant.
- The self and mutual inductances of the armature windings are a constant plus a second-harmonic sinusoidal function of the rotor position, with the amplitude of the second-harmonic component being the same for all self and mutual inductances.
- The mutual inductance between any winding on the field structure and any armature winding is a fundamental sinusoidal function of the rotor position.
- Effects of hysteresis are negligible or, in the case of cylindrical-rotor machines, are represented by the *g*-winding.

Basic equations

The electrical part of the machine is described by two set of equations.

1. The voltage equations:

$$\mathbf{V} = -\mathbf{RI} - \frac{d}{dt}[\mathbf{\Lambda}] \tag{5.1}$$

where

$$\mathbf{V} = \begin{bmatrix} V_a \\ V_b \\ V_c \\ -V_f \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.2}$$

$$\mathbf{R} = \begin{bmatrix} R_a & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & R_a & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & R_a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & R_f & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R_{kd} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & R_g & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{kq} \end{bmatrix} \tag{5.3}$$

$$\mathbf{I} = \begin{bmatrix} I_a \\ I_b \\ I_c \\ I_f \\ I_{kd} \\ I_g \\ I_{kq} \end{bmatrix} \tag{5.4}$$

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_a \\ \lambda_b \\ \lambda_c \\ \lambda_f \\ \lambda_{kd} \\ \lambda_g \\ \lambda_{kq} \end{bmatrix} \tag{5.5}$$

In the above equations, R_a is the armature resistance of each phase, $\mathbf{\Lambda}$ is the flux linkage vector, and the subscripts a, b, c, f, kd, g and kq represent the corresponding windings which have been defined previously.

2. The flux-current relationships and Park's transformation:

$$\mathbf{\Lambda} = \mathbf{LI} \tag{5.6}$$

All the windings of the machine are magnetically coupled with each other. The magnetic coupling between the windings is a function of the rotor position. Thus the flux linking each winding is also a function of the rotor position. This means that the instantaneous values of the elements of matrix **L** are functions of the rotor position. The coefficients of the resulting set of differential equations are periodic functions of the rotor angle and, therefore, functions of time. This time dependence can be avoided by transforming the armature quantities to a reference frame attached to the rotor. This is known as Park's transformation. New quantities are obtained from the projection of the actual variables on to three axes: one along the direct axis of the rotor field winding, called the direct axis, the second one along the neutral axis of the field winding, called the quadrature axis, and the third one on a stationary axis. The new quantities after transformation are completely uncoupled. The coefficients of the resulting equations are time-invariant. This transformation can be applied to the fluxes, currents as well as voltages. Thus, by definition,

$$\begin{bmatrix} \lambda_d \\ \lambda_q \\ \lambda_0 \end{bmatrix} = [\mathbf{P}^{-1}] \begin{bmatrix} \lambda_a \\ \lambda_b \\ \lambda_c \end{bmatrix} \tag{5.7}$$

where **P** is the Park's transformation matrix, and the subscripts d, q and 0 represent direct axis, quadrature axis and stationary axis respectively.

The Park's transformation matrix **P** is defined as:

$$[\mathbf{P}^{-1}] = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos\theta & \cos(\theta - 2\pi/3) & \cos(\theta + 2\pi/3) \\ \sin\theta & \sin(\theta - 2\pi/3) & \sin(\theta + 2\pi/3) \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \tag{5.8}$$

$$\theta = \omega_s t + \delta + \pi/2 \tag{5.9}$$

where θ is the d-axis rotor position in electrical radians with respect to a fixed reference position, δ is the synchronous torque angle and ω_s is the rated synchronous speed.

Similarly,

$$\begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix} = [\mathbf{P}^{-1}] \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{5.10}$$

$$\begin{bmatrix} I_d \\ I_q \\ I_0 \end{bmatrix} = [\mathbf{P}^{-1}] \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} \tag{5.11}$$

Since the rotor quantities remain unchanged, the transformation of variables yields the following set of voltage equations:

$$\begin{bmatrix} V_d \\ V_q \\ V_0 \\ -V_f \\ 0 \\ 0 \\ 0 \end{bmatrix} = - \begin{bmatrix} R_a & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & R_a & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & R_a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & R_f & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R_{kd} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & R_g & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{kq} \end{bmatrix} \begin{bmatrix} I_d \\ I_q \\ I_0 \\ I_f \\ I_{kd} \\ I_g \\ I_{kq} \end{bmatrix}$$

$$- \frac{d}{dt} \begin{bmatrix} \lambda_d \\ \lambda_q \\ \lambda_0 \\ \lambda_f \\ \lambda_{kd} \\ \lambda_g \\ \lambda_{kq} \end{bmatrix} + \begin{bmatrix} -\omega\lambda_q \\ \omega\lambda_d \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.12}$$

If the machine is Wye-connected with a neutral to ground impedance, $R_n + j\omega L_n$, then the following term must be added to the right-hand part of the zero-sequence equation:

$$3R_n I_0 + 3L_n \frac{dI_0}{dt}$$

The d-q-0 flux linkages are related to the d-q-0 currents by means of the following relationships:

$$\begin{bmatrix} \lambda_d \\ \lambda_f \\ \lambda_{kd} \end{bmatrix} = \begin{bmatrix} L_d & L_{a,f} & L_{a,kd} \\ L_{a,f} & L_f & L_{f,kd} \\ L_{a,kd} & L_{f,kd} & L_{kd} \end{bmatrix} \begin{bmatrix} I_d \\ I_f \\ I_{kd} \end{bmatrix} \tag{5.13}$$

$$\begin{bmatrix} \lambda_q \\ \lambda_g \\ \lambda_{kq} \end{bmatrix} = \begin{bmatrix} L_q & L_{a,g} & L_{a,kq} \\ L_{a,g} & L_g & L_{g,kq} \\ L_{a,kq} & L_{f,kq} & L_{kq} \end{bmatrix} \begin{bmatrix} I_q \\ I_g \\ I_{kq} \end{bmatrix} \tag{5.14}$$

$$\lambda_0 = L_0 I_0 \tag{5.15}$$

where L is the self or mutual inductances of the corresponding windings, and the subscript a corresponds to the armature winding. The coefficients of the above machine equations are known as the internal parameters of the machine. Since they cannot be obtained directly from measurement, they are usually calculated from the manufacturer's data [1] [6, 31].

5.2.2 Mechanical representation

Fig. 5.3 shows the mechanical part of the generator. With reference to the above figure, T_1, T_2 and T_3 respectively represent the input torques to the HP (High Pressure), the IP (Intermediate Pressure) and the LP (Low Pressure) turbines. T_{gen} is the electromagnetic torque of the generator and T_{exc} is the electromagnetic torque of the exciter. By using a single equivalent mass to represent the total effect produced by all the masses, the speed variation of the machine can be modelled with the following equations:

$$J\frac{d\omega}{dt} + D\omega = T_{tur} - T_{gen} \tag{5.16}$$

$$\frac{d\theta}{dt} = \omega \tag{5.17}$$

$$\frac{d\delta}{dt} = \omega - \omega_s \tag{5.18}$$

[1] The following is a typical set of manufacturer's machine parameters which can be obtained from standardized machine tests: armature resistance (R_a), armature leakage reactance (X_l), zero-sequence reactance (X_0), transient reactances ($X_d^{'}$, $X_q^{'}$), subtransient reactances ($X_d^{"}$, $X_q^{"}$), transient open-circuit time constants ($\tau_{do}^{'}$, $\tau_{qo}^{'}$) and subtransient open-circuit time constants ($\tau_{do}^{"}$, $\tau_{qo}^{"}$).

where J is the moment of inertia, D is the damping coefficient and, T_{tur} is the total equivalent torque applied to the turbines. The inertia constant H (seconds) is related to the moment of inertia J and machine rating S_{rating} by the following equation:

$$H = \frac{J\omega_s^2/2}{S_{rating}} \tag{5.19}$$

The electrical and mechanical equations are related by the rotor position and the electromagnetic torques:

$$\theta_{mech} = \frac{\theta_{elec}}{N_p/2} \tag{5.20}$$

$$T_{gen} = \frac{N_p(\lambda_d I_q - \lambda_q I_d)}{2} \tag{5.21}$$

where N_p is the number of poles.

5.3 AVR control system

In modern power systems, generators have an automatic way of controlling the voltages at their terminals. Changes in the electric system conditions, such as variations on load or faults that are cleared by tripping lines and/or load, produce a severe change in the machine voltage that can be controlled by changing the field voltage of the generator exciter. This change is made automatically by a control circuit that measures the terminal voltage and compares it against a preset value. If there is a difference, the field voltage is changed to keep the terminal voltage constant. An IEEE AVR type-1 excitation system [34] as shown in fig. 5.4 has been implemented to control the voltage output of the generator. Symbol definitions are as defined in [34]. The values for the gains and time constants are tabulated in table 5.1 [31]. The saturation function S_E is modelled according to

the following formula:

$$S_E = f(E_{FD}) = A_{ex} e^{B_{ex} E_{FD}} \tag{5.22}$$

where

$$A_{ex} = \frac{S^4_{E.75MAX}}{S^3_{E_{MAX}}} \tag{5.23}$$

$$B_{ex} = \frac{4}{E_{FD_{MAX}}} \ln \frac{S_{E_{MAX}}}{S_{E.75MAX}} \tag{5.24}$$

Symbol	Self-excited Commutator with Amplidyne voltage regulator	Self-excited commutator exciter with Mag-A-Stat voltage regulator	Rotating rectifier exciter with static voltage regulator
T_R	0.0 to 0.06	0.0	0.0
K_A	25 To 50	400	400
T_A	0.06 to 0.2	0.05	0.02
$V_{R_{MAX}}$	1.0	3.5	7.3
$V_{R_{MIN}}$	-1.0	-3.5	-7.3
K_F	0.01 to 0.08	0.04	0.03
T_F	0.35 to 1.0	1.0	1.0
K_E	-0.05	-0.17	1.0
T_E	0.5	0.95	0.8
$S_{E_{MAX}}$	0.267	0.95	0.86
$S_{E_{0.75MAX}}$	0.074	0.22	0.50

Table 5.1: Typical constants of excitation systems on 3600 rpm steam turbine generators.

5.4 Governor control system

Governor is used to regulate the input power to the generator by controlling the valves of the turbine, trying to maintain a constant speed in the shaft. An IEEE steam turbine system [35] as shown in fig. 5.5 has been implemented to simulate

the governor and turbine control. Symbol definitions are as defined in [35]. It should be mentioned that only the HP turbine has any important effect on the response of the system due to its relatively small time constants. The IP and LP turbines have large time constants and have little effect to the electromechanical transients under studies. Table 5.2 shows typical values for the gains and time constants used for fossil steam turbines.

Symbol	Value
P_{MAX} (pu)	1.0
P_{MIN} (pu)	0.0
T_1 (s)	0.134
T_2 (s)	0.0
T_3 (s)	0.231
T_4 (s)	0.135
K_1 (pu)	1.0

Table 5.2: Typical parameters for IEEE speed steam governing system.

5.5 Generator transformer

In order to minimize the transmission loss, the machine terminal voltage must be increased from generation level to transmission level. This is done by a step up transformer. Fig. 5.6 shows a three phase delta-star connected transformer model which is used in the simulator. It has been assumed that the transformer has low homopolar reluctance. In this case, an iron-core path is provided for the return of the zero-sequence flux. It is also assumed that the magnetic induction of the three-phases is independent and the zero sequence parameters are the same as the positive sequence parameters. This implies that only the saturation curve for one core leg is needed. Thus, the transformer can be modelled by using three separate, identical single-phase units. Referring to fig. 5.6, R_1 and L_1 are the

primary winding leakage-impedance parameters. R_2 and L_2 are the secondary leakage-impedance parameters. They are the same for all three phases. These values can be obtained from short-circuit tests [2]:

$$R_{short} = \frac{P_{short,loss}}{I_{short}^2} \tag{5.25}$$

$$X_{short} = \frac{1}{I_{short}}\sqrt{U_{short}^2 - \frac{P_{short,loss}^2}{I_{short}^2}} \tag{5.26}$$

$$R_1 = \frac{R_{short}U_{lv}^2}{2U_{hv}^2} \tag{5.27}$$

$$X_1 = \frac{X_{short}U_{lv}^2}{2U_{hv}^2} \tag{5.28}$$

$$L_1 = \frac{X_1}{\omega_s} \tag{5.29}$$

$$R_2 = \frac{R_{short}}{2} \tag{5.30}$$

$$X_2 = \frac{X_{short}}{2} \tag{5.31}$$

$$L_2 = \frac{X_2}{\omega_s} \tag{5.32}$$

where U_{short} is the applied primary voltage, U_{lv} is the rated primary voltage and U_{hv} is the rated secondary voltage. The resistance, R_{mag}, which takes account for the excitation core losses, is calculated as:

$$R_{mag} = \frac{U_{lv}^2}{P_{ex,loss}} \tag{5.33}$$

where $P_{ex,loss}$ is the excitation losses.

[2] In a short circuit test, the secondary is short-circuited and, a low voltage (U_{short}) is applied to the primary so that full-load currents (I_{short}) are circulated in the secondary.

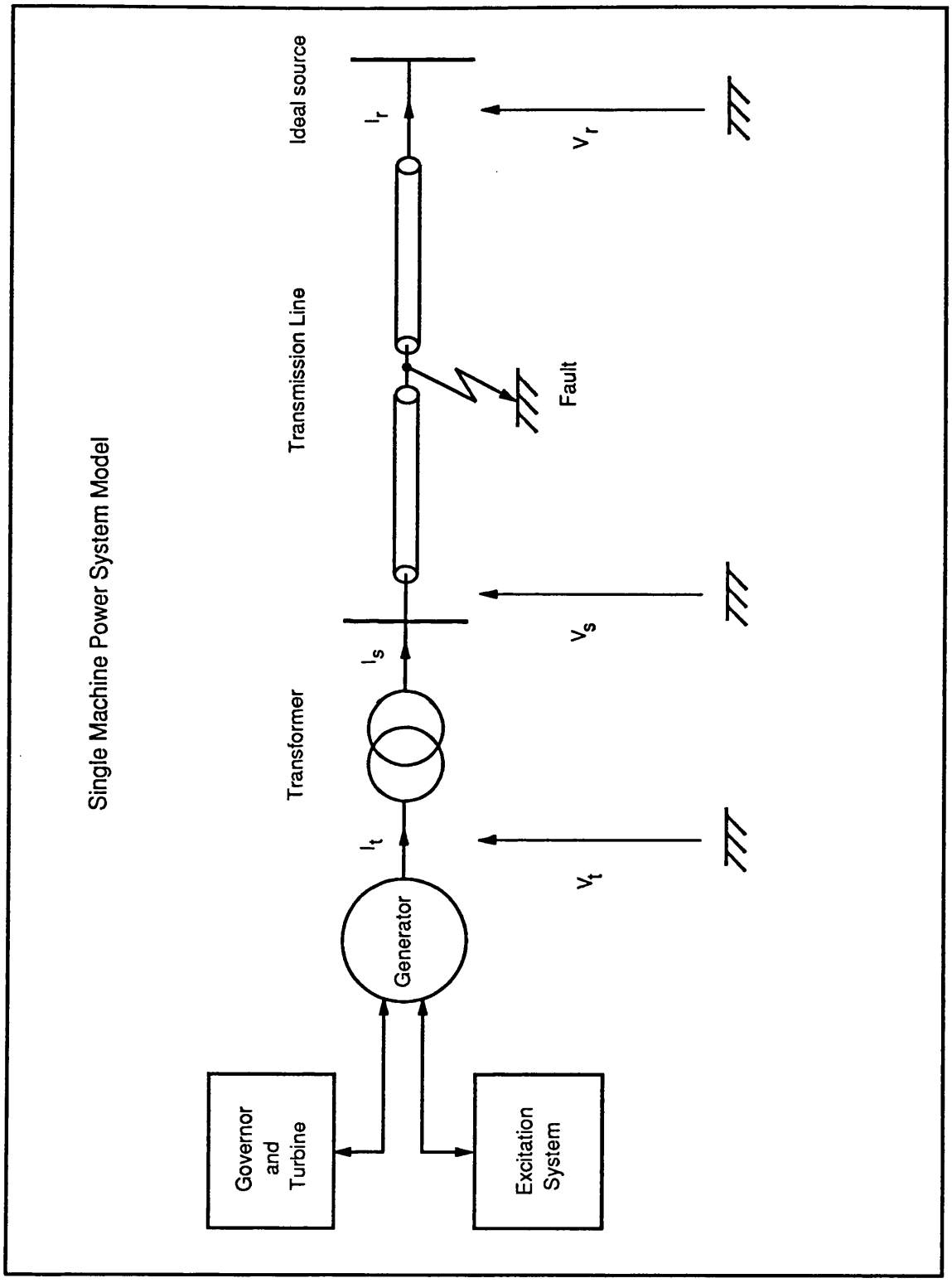

Figure 5.1: Single machine power system

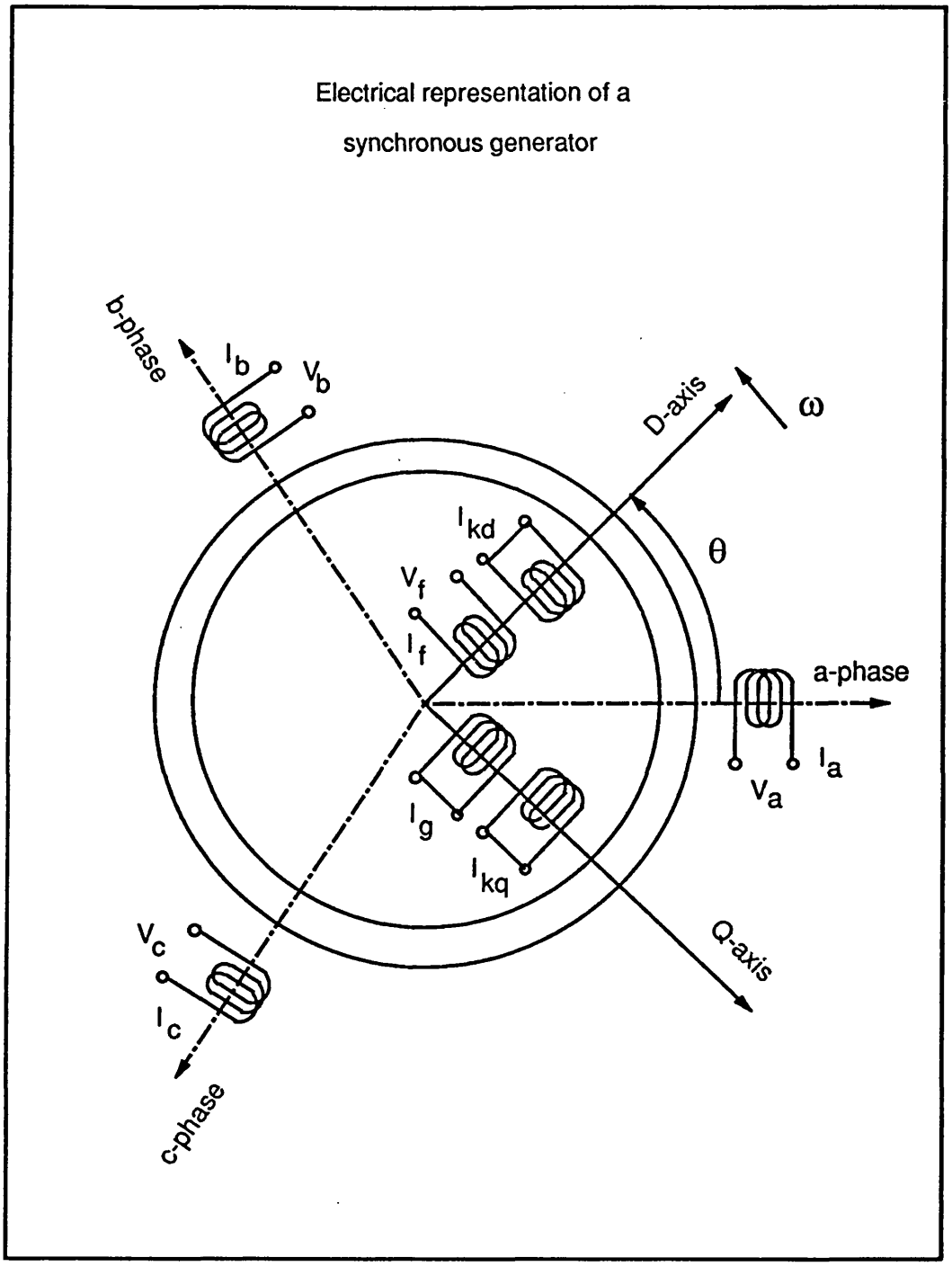

Figure 5.2: Electrical part of a synchronous generator

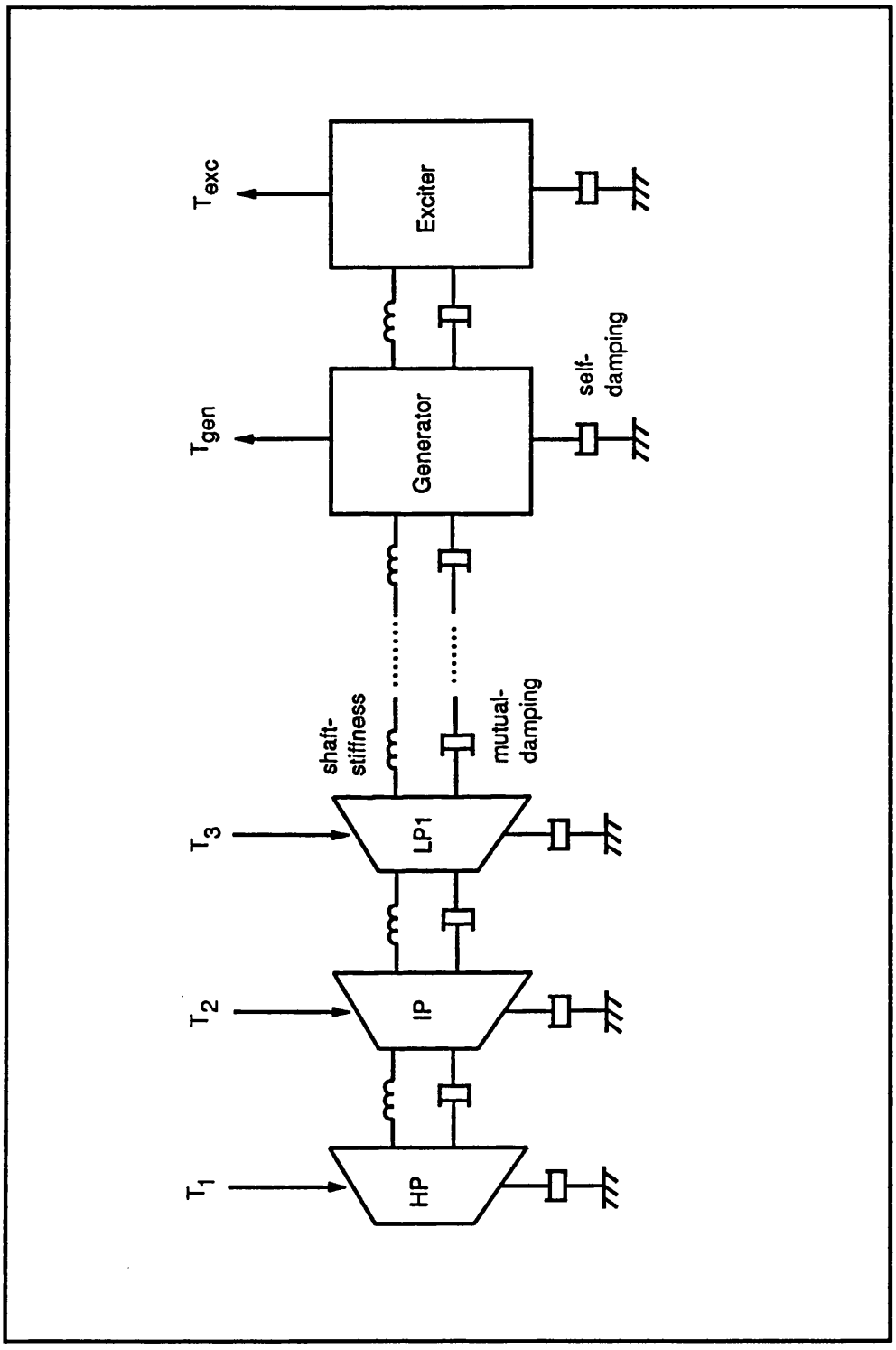

Figure 5.3: Mechanical part of a synchronous generator

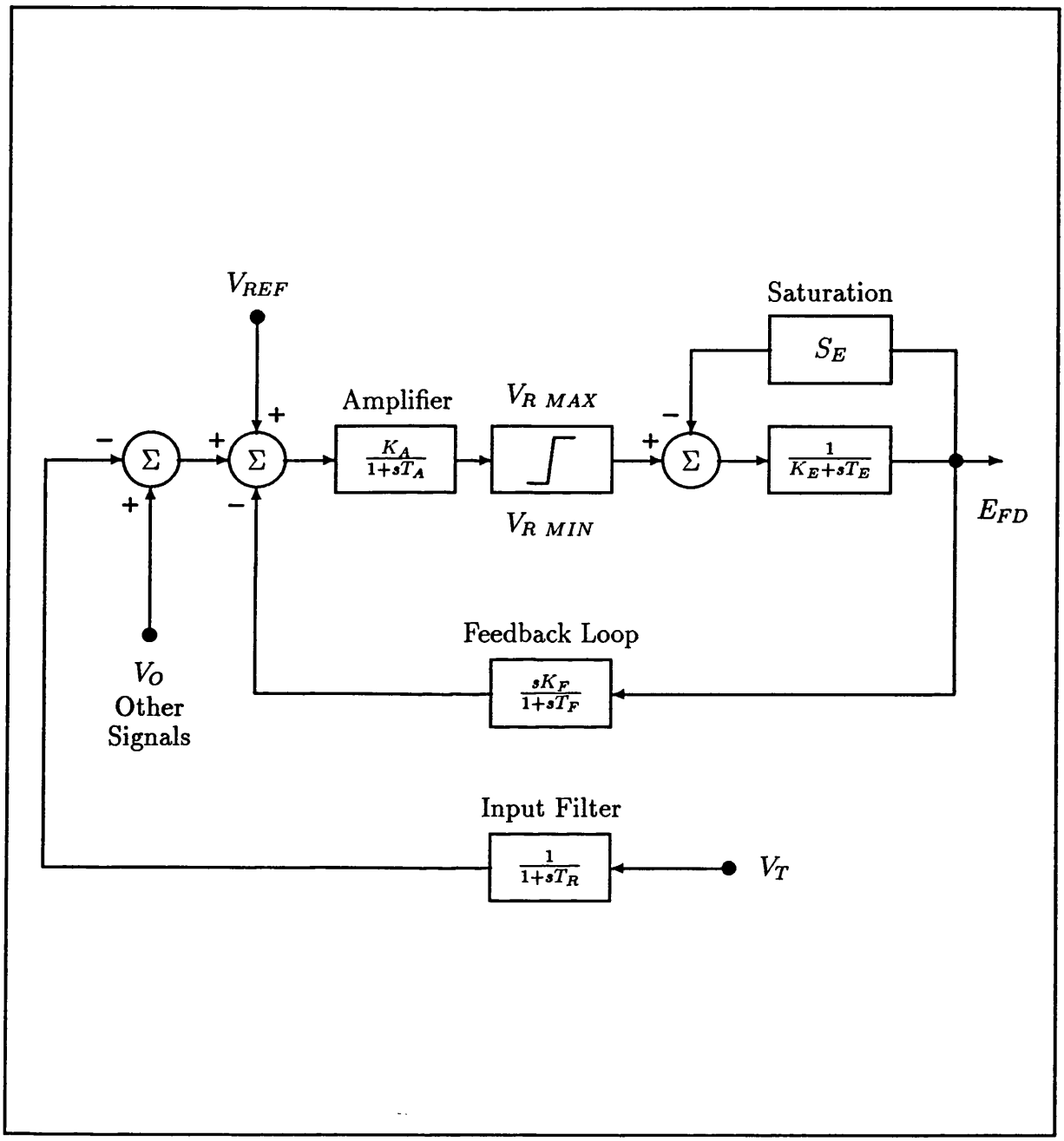

Figure 5.4: IEEE type-1 AVR excitation control

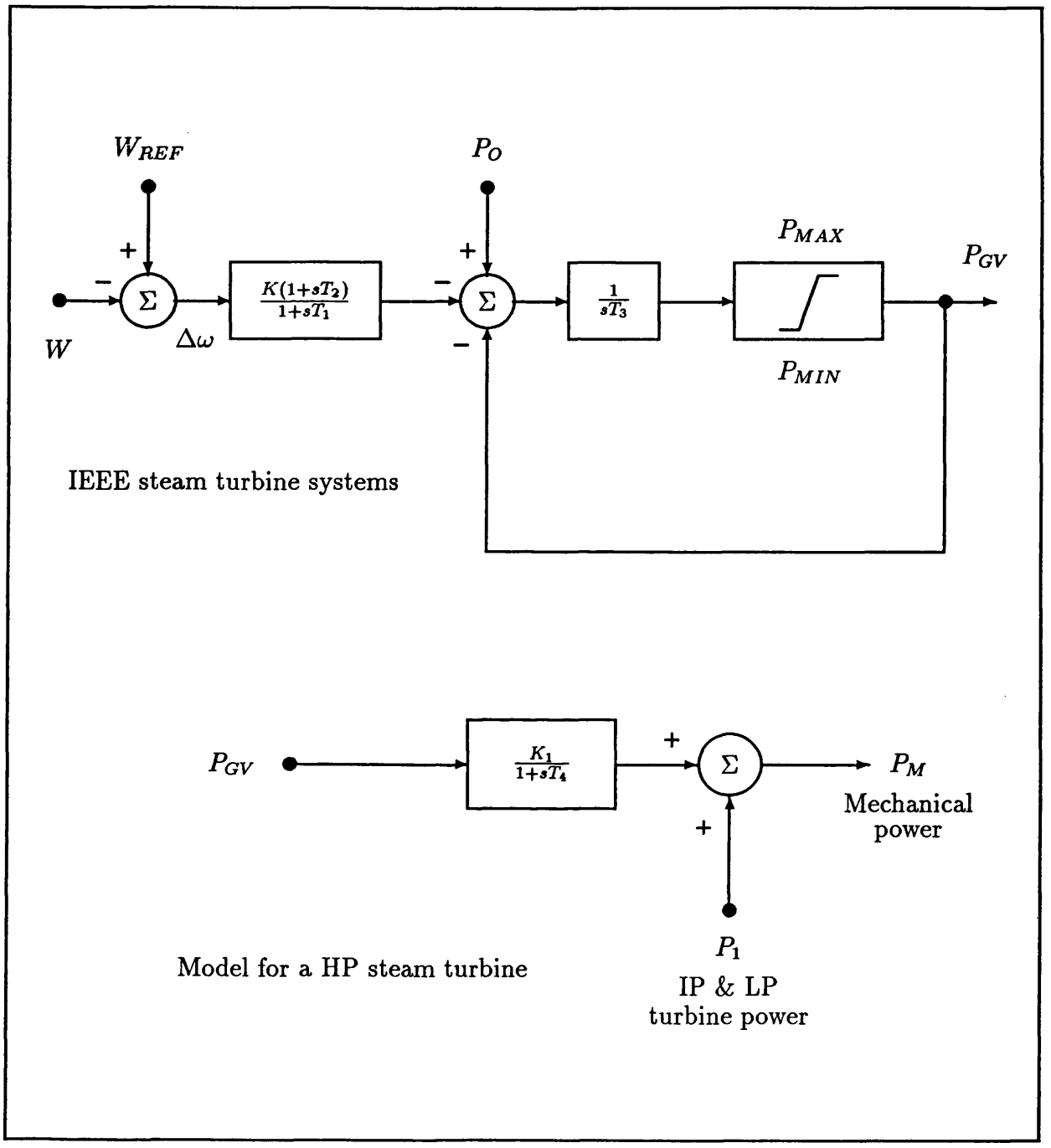

Figure 5.5: IEEE steam governor and turbine systems

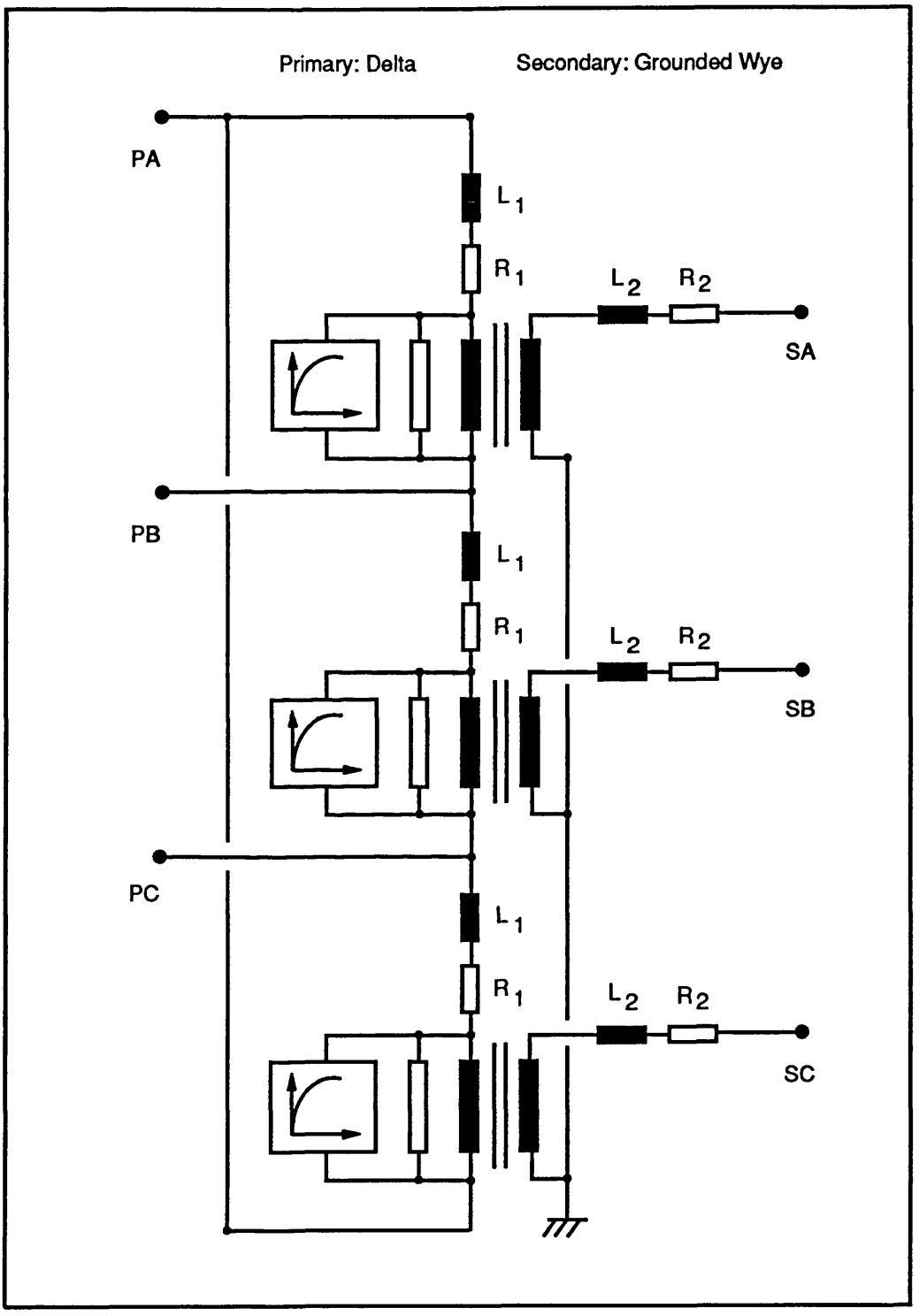

Figure 5.6: Generator transformer model

Chapter 6

Simulation techniques

6.1 Introduction

Chapter 4 describes the electromagnetic transient models based on the frequency domain and various time domain approaches. Chapter 5 describes a detailed power system source model for studying the low frequency electromechanical transients. In this chapter, the numerical calculation techniques used by the time-convolution, frequency domain and EMTP techniques for power system fault transient modelling are described. In order to incorporate the frequency dependence of the system parameters, the frequency response of the system must be evaluated. In this respect, Fourier transform and inverse Fourier transform techniques, whereby the time domain data can be transformed into and out of frequency domain, have been proven to be invaluable. The time domain simulation of the transmission line is based on Bergeron's travelling wave model [8]. It uses linear relationships between current and voltage which are invariant from the point of view of an observer travelling with the wave. Techniques have been developed so that any type of fault sequence occurring at any point on the transmission line can be simulated.

6.2 Mathematical theory overview

Convolution: The convolution of two given functions $f_1(t)$ and $f_2(t)$ is defined to be the following function:

$$f(t) = \int_{-\infty}^{\infty} f_1(x) f_2(t-x) dx = f_1(t) * f_2(t) \tag{6.1}$$

Fourier transform: The Fourier transform of $f(t)$ (symbolized by $\mathcal{F}$) is defined by:

$$F(\omega) = \mathcal{F}[\ f(t)\ ] = \int_{-\infty}^{\infty} f(t) exp(-j\omega t) dt \tag{6.2}$$

Inverse Fourier transform: The Inverse Fourier transform of $F(\omega)$ (symbolized by $\mathcal{F}^{-1}$) is defined by:

$$f(t) = \mathcal{F}^{-1}[\ F(\omega)\ ] = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) exp(j\omega t) d\omega \tag{6.3}$$

Time convolution theorem: The time convolution theorem states that if

$$\mathcal{F}[\ f_1(t)\ ] = F_1(\omega)$$

$$\mathcal{F}[\ f_2(t)\ ] = F_2(\omega)$$

then

$$\mathcal{F}[\ f_1(t)\ *\ f_2(t)\ ] = F_1(\omega)\ F_2(\omega) \tag{6.4}$$

or

$$\mathcal{F}^{-1}\{\ F_1(\omega)\ F_2(\omega)\ \} =\ f_1(t)\ *\ f_2(t) \tag{6.5}$$

Unit impulse response: The response of a linear system to a unit impulse $\delta(t)$ is called the unit impulse response, $u(t)$, of the system.

Response of a time-invariant system to any arbitrary input, $g(t)$ is then given by:

$$f(t) = \int_{-\infty}^{\infty} g(\tau) u(t-\tau) d\tau = g(t) * u(t) \tag{6.6}$$

Eqn. 6.6 indicates a very interesting result: the response of a linear system is uniquely determined from the knowledge of the unit impulse response, $u(t)$ of the system.

6.3 Time convolutional technique

The time convolutional technique refers specifically to the process by which numerical solutions for transmission line impulse response functions, as formed from inverse Fourier transform evaluations, are used in time-convolution. Essentially, there are two stages of analysis. The first is a preparatory phase in which impulse response functions are formed. This arises only once for any given transmission system, following which the impulse responses found for it are held with other relevant transmission line data. The second stage is the one in which electromagnetic transient analysis is carried out and which requires evaluations of time convolutional integrals.

6.3.1 Numerical approximations

The solution of the convolution integral shown in eqn. 6.6 is essentially a matter of solving an integration numerically. The following numerical technique, which is suggested by N.J. LOY [36] for designing a FIR digital filter, is used. Basically, since a distributed model of an EHV transmission system consists of a series of cascaded LRC circuits, it can effectively be treated as a digital filter of order n and the theory of digital filter design [36] can be applied and used for the simulation studies.

$$ANS(n) = \sum_{k=0}^{n-1} Coef(k) \times Data(n-k) \tag{6.7}$$

for $n = 0, 1, \cdots, 2n - 2$.

where $Coef(k)$ is the impulse response samples of the system and
$Data(n - k)$ is the input samples to the system.

Diagrammatically, this can be represented as shown in fig. 6.1. An output point is yielded by summing all the multiplication results, which are obtained by multiplying together the data points and their corresponding coefficients. The data points are then shifted with respect to the coefficients. One new datum point is entered into the string while the oldest datum point is lost. The multiplication-summation is then repeated to generate the next output point.

6.3.2 Fault transient modelling of single conductor line

The fault transient model of a single conductor line with an earth return (fig. 4.1) is represented as shown in fig. 4.3. Chapter 4.5 derives the universal time domain relationships which relate sending end, receiving end and fault point parameters. These relationships are as shown in equations 4.65 and 4.66 and for the sake of convenience, they are restated as shown below:

$$\begin{bmatrix} i_f(t) \\ i_s(t) \\ i_r(t) \end{bmatrix} = \begin{bmatrix} y_a & y_b & y_c \\ y_d & y_e & y_f \\ y_g & y_h & y_i \end{bmatrix} * \begin{bmatrix} e_{ff}(t) \\ e_s(t) \\ e_r(t) \end{bmatrix} \tag{6.8}$$

$$\begin{bmatrix} e_{ff}(t) \\ e_s(t) \\ e_r(t) \end{bmatrix} = \begin{bmatrix} z_a & z_b & z_c \\ z_d & z_e & z_f \\ z_g & z_h & z_i \end{bmatrix} * \begin{bmatrix} i_f(t) \\ i_s(t) \\ i_r(t) \end{bmatrix} \tag{6.9}$$

All the transient calculations are based on these two sets of universal equations. The impulse response of each element of the $[z]$-matrix and $[y]$-matrix in equations 6.8 and 6.9 is precalculated and stored before starting the fault transient calculations.

6.3.3 Fault sequences

The following is a typical fault sequence of events, which will occur after the fault inception, for a double end fed transmission system:

Event 0 : Pre-fault steady state conditions.

Event 1 : A fault to ground occurs at the fault point on the line at time t_{fault}.

Event 2 : The circuit breaker poles of both the receiving end and the sending end open simultaneously at time t_{open}.

Event 3 : The fault is cleared at time t_{clear}.

Event 4 : The circuit breaker poles of both the receiving end and the sending end reclose simultaneously at time $t_{reclose}$.

It has been assumed in the simulation that the circuit breaker poles at both ends are opened *simultaneously* at time t_{open} and are reclosed simultaneously at time $t_{reclose}$. Figs. 2.3 and 2.4 illustrate typical sequence of autoreclose operations for oil and air circuit breakers. The dead time which can be attained with and without using special high speed reclosing mechanism is also shown in each figure.

6.3.4 Fault transient calculation

Event 1: the fault inception transients

The transient model for simulating this event is shown in fig. 6.2. In order to simulate the fault inception, a forcing voltage e_{ft1}, which opposes the prefault

steady state voltage v_{fs}(t), is injected to the system at the fault point. This can be represented mathematically by the following equation:

$$e_{fft1} = -v_{fs}(t)h(t - t_{fault}) \tag{6.10}$$

Since the circuit breakers are still closed during this event, the following conditions are satisfied:

$$e_{st1} = e_{rt1} = 0 \tag{6.11}$$

By solving the system equation 6.8, i_{ft1}, i_{st1} and i_{rt1} can thus be obtained, i.e.,

$$i_{ft1} = y_a * e_{fft1} \tag{6.12}$$

$$i_{st1} = y_d * e_{fft1} \tag{6.13}$$

$$i_{rt1} = y_g * e_{fft1} \tag{6.14}$$

The voltages, v_{st1}, v_{rt1} and v_{ft1} can be solved using the following equations:

$$v_{st1} = -z_{sd} * e_{fft1} \tag{6.15}$$

$$v_{rt1} = -z_{rg} * e_{fft1} \tag{6.16}$$

$$v_{ft1} = e_{fft1} + z_f * i_{ft1} \tag{6.17}$$

where z_{sd} is the impulse response of $Z_{ss}Y_D$, and z_{rg} is the impulse response of $Z_{sr}Y_G$. Z_{ss} and Z_{sr} are the sending end and receiving end source impedances respectively. z_f is the impulse response of the fault path impedance (Z_f).

The transients calculated are then added to the corresponding steady state values to obtain the resultant system fault inception responses.

Events 2: the breaker pole opening transients

The transient model for simulating this event is shown in fig. 6.3. In order to simulate the opening of the circuit breaker poles, currents i_{st2} and i_{rt2} are forced

to oppose the currents i_s and i_r respectively at time t_{open}. This can be represented mathematically by the following equations:

$$i_{st2} = -i_s h(t - t_{open}) \tag{6.18}$$

$$i_{rt2} = -i_r h(t - t_{open}) \tag{6.19}$$

Since $e_{fft2} = 0$, eqn. 6.9 gives:

$$z_a * i_{ft2} + z_b * i_{st2} + z_c * i_{rt2} = 0 \tag{6.20}$$

The corresponding frequency domain expression of this equation is:

$$Z_A I_{ft2} + Z_B I_{st2} + Z_C I_{rt2} = 0 \tag{6.21}$$

This gives:

$$I_{ft2} = -Z_A^{-1} Z_B I_{st2} - Z_A^{-1} Z_C I_{rt2} \tag{6.22}$$

From eqn. 6.22, i_{ft2} is then calculated as:

$$i_{ft2} = -z_{ba} * i_{st2} - z_{ca} * i_{rt2} \tag{6.23}$$

where z_{ba} is the impulse response of $\left[Z_A^{-1} Z_B\right]$, and z_{ca} is the impulse response of $\left[Z_A^{-1} Z_C\right]$.

e_{st2} and e_{rt2} can now be obtained from system equation 6.9 and the voltages, v_{st2}, v_{rt2} and v_{ft2} are found using the following equations:

$$v_{st2} = e_{st2} - z_{ss} * i_{st2} \tag{6.24}$$

$$v_{rt2} = e_{rt2} - z_{sr} * i_{rt2} \tag{6.25}$$

$$v_{ft2} = z_f * i_{ft2} \tag{6.26}$$

$$e_{fft2} = 0 \tag{6.27}$$

where z_{ss} and z_{sr} are the impulse response of the sending end and receiving end source impedances (Z_{ss} and Z_{sr}) respectively.

The transients calculated are then added to the corresponding steady state values to obtain the resultant system breaker pole opening responses.

Event 3: the fault clearing transients

The transient model for simulating this event is shown in fig. 6.4. In order to simulate the fault clearance transient, current i_{ft3} is forced to oppose fault current i_f at time t_{clear}. This can be represented mathematically by the following equation:

$$i_{ft3} = -i_f h(t - t_{clear}) \tag{6.28}$$

Since the circuit breakers are now open during this event, the following conditions are satisfied:

$$i_{st3} = i_{rt3} = 0 \tag{6.29}$$

Solving the system equation 6.9, e_{fft3}, e_{st3} and e_{rt3} can be obtained and the voltages, v_{st3}, v_{rt3} and v_{ft3} are found using the following identities:

$$v_{st3} = e_{st3} \tag{6.30}$$

$$v_{rt3} = e_{rt3} \tag{6.31}$$

$$v_{ft3} = z_f * i_{ft3} \tag{6.32}$$

These transients are then added to the corresponding steady state values to obtain the resultant system fault clearance responses.

Event 4: the breaker pole reclosing transients

The transient model for simulating this event is shown in fig. 6.5. In order to simulate the reclosure transient of the circuit breaker poles, the voltages e_{st4}

and e_{rt4} are forced to oppose e_s and e_r respectively at time $t_{reclose}$. This can be represented mathematically by the following equations:

$$e_{st4} = -e_s h(t - t_{reclose}) \tag{6.33}$$

$$e_{rt4} = -e_r h(t - t_{reclose}) \tag{6.34}$$

Since $i_{ft4} = 0$, eqn. 6.8 gives:

$$y_a * e_{fft4} + y_b * e_{st4} + y_c * e_{rt4} = 0 \tag{6.35}$$

The corresponding frequency domain expression of this equation is:

$$Y_A E_{fft4} + Y_B E_{st4} + Y_C E_{rt4} = 0 \tag{6.36}$$

This gives:

$$E_{fft4} = -Y_A^{-1} Y_B E_{st4} - Y_A^{-1} Y_C E_{rt4} \tag{6.37}$$

From eqn. 6.37, e_{fft4} is then calculated as:

$$e_{fft4} = -y_{ba} * e_{st4} - y_{ca} * e_{rt4} \tag{6.38}$$

where y_{ba} is the impulse response of $\left[Y_A^{-1} Y_B\right]$, and y_{ca} is the impulse response of $\left[Y_A^{-1} Y_C\right]$.

i_{st4} and i_{rt4} can now be obtained by solving the system equation 6.8 and the voltages, v_{st4}, v_{rt4} and v_{ft4} are obtained by solving the following equations:

$$v_{st4} = e_{st4} - z_{ss} * i_{st4} \tag{6.39}$$

$$v_{rt4} = e_{rt4} - z_{sr} * i_{rt4} \tag{6.40}$$

$$v_{ft4} = e_{fft4} \tag{6.41}$$

These transients are then added to the corresponding steady state values to obtain the resultant system breaker pole reclosure responses.

6.4 Frequency domain technique

The model for processing sampled signal data involves converting from the time-domain representation to the frequency-domain representation. This is so because it is more convenient, for both analog and digital systems, to process signals in the frequency-domain.

The Fourier transform and the inverse Fourier transform as shown in equations 6.2 and 6.3 are very powerful tools by which the frequency-domain functions can be transformed into the corresponding time-domain functions and vice versa. Several authors [37, 38] have developed and successfully used a modified half-range form of the basic Fourier integral as given in eqn. 6.42.

$$f(t) = Re\frac{exp(at)}{\pi}\int_0^{\Omega} \sigma f(\omega - ja)exp(j\omega t)d\omega \tag{6.42}$$

The finite range of integration gives rise to Gibbs oscillations, and this is overcome by introducing the sigma factor given in eqn. 6.43.

$$\sigma = \frac{sin(\pi\omega/\Omega)}{(\pi\omega/\Omega)} \tag{6.43}$$

In addition, a frequency shift constant a is introduced to ensure numerical stability when the integral is evaluated digitally. The relationship between the sigma factor, frequency shift constant, and truncation frequency, is the subject of an extensive study by Day *et al* [37].

The Discrete Fourier Transform (DFT) converts samples of a time-domain signal to samples of the different signal frequencies and their amplitudes. To perform the conversion, the DFT calculation requires n^2 complex multiplications and additions; where n equals the number of samples and is usually a power of two.

The Fast Fourier Transform (FFT) [39] is an elegant implementation of the DFT, which reduces the number of complex multiplications and additions required to $n \log_2 n$. That is more than 100 times fewer for a 1024-point FFT. The FFT algorithm used also has the advantage of not requiring data storage beyond the original data samples. Detailed FFT algorithm, which is used in the project, is given in Appendix C.

6.4.1 Frequency domain modelling

6.4.2 Fault transient modelling of single conductor line

The fault transient model of a single conductor transmission system is shown in fig. 4.3. Chapter 4.2 derives the universal frequency domain relationships which relate sending end, receiving end and fault point parameters. These relationships are as shown in equations 4.21 and 4.22. All the transient calculations are based on these two sets of universal equations. The frequency spectrum of each element of the $[Y]$-matrix and $[Z]$-matrix in equations 4.21 and 4.22 is precalculated and stored before starting the fault transient calculations.

6.4.3 Fault sequences

The same fault sequence of events as described in section 6.3.3 for time convolutional modelling of a single conductor system is also used for frequency domain modelling.

6.4.4 Fault transient calculation

For a single conductor system (fig. 4.1), similar fault transient calculations as those described in section 6.3.4 for each event are used for the frequency domain modelling. However, some differences must be noted for each event. Firstly, all the time-domain data must be transformed into frequency domain data using the FFT routine. It must also be remembered that convolutions in the time-domain are multiplications in the frequency domain. The transient calculations are performed in the frequency-domain by solving the universal matrix equations 4.21 and 4.22. After that, the results are transformed back into the time-domain and superimposed on the steady-state values of the previous event.

6.4.5 Fault transient modelling of three phase line

For a three phase system (fig. 4.2), each element of the universal matrix in equations 4.21 and 4.22 is a 3 by 3 submatrix. Thus, the following relationships, which relate the sending end, the receiving end and the fault point parameters, can be

obtained:

$$\begin{bmatrix} I_{fa} \\ I_{fb} \\ I_{fc} \\ I_{sa} \\ I_{sb} \\ I_{sc} \\ I_{ra} \\ I_{rb} \\ I_{rc} \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & Y_{14} & Y_{15} & Y_{16} & Y_{17} & Y_{18} & Y_{19} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} & Y_{25} & Y_{26} & Y_{27} & Y_{28} & Y_{29} \\ Y_{31} & Y_{32} & Y_{33} & Y_{34} & Y_{35} & Y_{36} & Y_{37} & Y_{38} & Y_{39} \\ Y_{41} & Y_{42} & Y_{43} & Y_{44} & Y_{45} & Y_{46} & Y_{47} & Y_{48} & Y_{49} \\ Y_{51} & Y_{52} & Y_{53} & Y_{54} & Y_{55} & Y_{56} & Y_{57} & Y_{58} & Y_{59} \\ Y_{61} & Y_{62} & Y_{63} & Y_{64} & Y_{65} & Y_{66} & Y_{67} & Y_{68} & Y_{69} \\ Y_{71} & Y_{72} & Y_{73} & Y_{74} & Y_{75} & Y_{76} & Y_{77} & Y_{78} & Y_{79} \\ Y_{81} & Y_{82} & Y_{83} & Y_{84} & Y_{85} & Y_{86} & Y_{87} & Y_{88} & Y_{89} \\ Y_{91} & Y_{92} & Y_{93} & Y_{94} & Y_{95} & Y_{96} & Y_{97} & Y_{98} & Y_{99} \end{bmatrix} \begin{bmatrix} E_{ffa} \\ E_{ffb} \\ E_{ffc} \\ E_{sa} \\ E_{sb} \\ E_{sc} \\ E_{ra} \\ E_{rb} \\ E_{rc} \end{bmatrix} \tag{6.44}$$

$$\begin{bmatrix} E_{ffa} \\ E_{ffb} \\ E_{ffc} \\ E_{sa} \\ E_{sb} \\ E_{sc} \\ E_{ra} \\ E_{rb} \\ E_{rc} \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} & Z_{15} & Z_{16} & Z_{17} & Z_{18} & Z_{19} \\ Z_{21} & Z_{22} & Z_{23} & Z_{24} & Z_{25} & Z_{26} & Z_{27} & Z_{28} & Z_{29} \\ Z_{31} & Z_{32} & Z_{33} & Z_{34} & Z_{35} & Z_{36} & Z_{37} & Z_{38} & Z_{39} \\ Z_{41} & Z_{42} & Z_{43} & Z_{44} & Z_{45} & Z_{46} & Z_{47} & Z_{48} & Z_{49} \\ Z_{51} & Z_{52} & Z_{53} & Z_{54} & Z_{55} & Z_{56} & Z_{57} & Z_{58} & Z_{59} \\ Z_{61} & Z_{62} & Z_{63} & Z_{64} & Z_{65} & Z_{66} & Z_{67} & Z_{68} & Z_{69} \\ Z_{71} & Z_{72} & Z_{73} & Z_{74} & Z_{75} & Z_{76} & Z_{77} & Z_{78} & Z_{79} \\ Z_{81} & Z_{82} & Z_{83} & Z_{84} & Z_{85} & Z_{86} & Z_{87} & Z_{88} & Z_{89} \\ Z_{91} & Z_{92} & Z_{93} & Z_{94} & Z_{95} & Z_{96} & Z_{97} & Z_{98} & Z_{99} \end{bmatrix} \begin{bmatrix} I_{fa} \\ I_{fb} \\ I_{fc} \\ I_{sa} \\ I_{sb} \\ I_{sc} \\ I_{ra} \\ I_{rb} \\ I_{rc} \end{bmatrix} \tag{6.45}$$

where subscripts a, b and c correspond to phase-'a', phase-'b' and phase-'c' respectively. Detailed definition of each element of the above two matrix equations is given in Appendix B.

Following similar procedures as described in section 6.4.2 for a single conductor line system, the fault transients of a three phase system can be obtained. However, two major differences should be noted. Firstly, there are different fault types, such as single phase to earth, double phase to earth, three phase to earth and pure interphase faults for a three phase system. Secondly, when doing the frequency domain transient calculations, it is necessary to select appropriate submatrices from the universal system matrices as shown in eqn. 6.44 and eqn. 6.45 according to the fault type and solving the selected equations for the appropriate system parameters for each event of the autoreclosure sequence.

6.5 EMTP simulation technique

Power system plant components, other than transmission lines, can be simulated by equivalent circuits consisting of combinations of voltage and current sources, resistances, inductances and capacitances. The transmission line 'component' has to be treated separately. This is so because it is an extremely difficult and complicated matter to incorporate both the frequency dependent parameters and the distributed losses of the transmission line in the time domain. Section 4.4 elaborates the modelling of distributed frequency dependent transmission line in the time domain. This section describes the basic numerical calculation techniques used in the simulation of the overall system.

6.5.1 Simulation of basic components

Fig. 6.6 shows the equivalent circuits for simulating three basic power system components, namely, resistance (R), inductance (L) and capacitance (C).

- **Resistance**: Referring to fig. 6.6a, the following equation can be used to simulate a resistance:

$$v_k(t) - v_m(t) = Ri_{k,m}(t) \tag{6.46}$$

or

$$i_{k,m}(t) = \frac{1}{R}[v_k(t) - v_m(t)] \tag{6.47}$$

- **Inductance**: Referring to fig. 6.6b(i), the following equation can be used to simulate an inductance:

$$v_k(t) - v_m(t) = L\frac{di_{k,m}}{dt} \tag{6.48}$$

Eqn. 6.48 can be solved numerically using trapezoidal rule as follows:

$$\begin{aligned} i_{k,m}(t) &= i_{k,m}(t-\Delta t) + \frac{1}{L}\int_{t-\Delta t}^{t}(v_k - v_m)dt \\ &= i_{k,m}(t-\Delta t) + \frac{1}{L}\frac{\Delta t}{2}\{[v_k(t) - v_m(t)] + [v_k(t-\Delta t) - v_m(t-\Delta t)]\} \\ &= I_{k,m}(t-\Delta t) + \frac{\Delta t}{2L}[v_k(t) - v_m(t)] \end{aligned} \tag{6.49}$$

where

$$I_{k,m}(t-\Delta t) = i_{k,m}(t-\Delta t) + \frac{\Delta t}{2L}[v_k(t-\Delta t) - v_m(t-\Delta t)] \tag{6.50}$$

This can be visualised as shown in fig. 6.6b(ii).

- **Capacitance**: Referring to fig. 6.6c(i), the following equation can be used to simulate a capacitance:

$$i_{k,m}(t) = C\frac{d}{dt}[v_k(t) - v_m(t)] \tag{6.51}$$

The above equation can be solved numerically using trapezoidal rule as follows:

$$\begin{aligned} v_k(t) - v_m(t) &= v_k(t-\Delta t) - v_m(t-\Delta t) + \frac{1}{C}\int_{t-\Delta t}^{t} i_{k,m}(t)dt \\ i_{k,m}(t) &= I_{k,m}(t-\Delta t) + \frac{2C}{\Delta t}[v_k(t) - v_m(t)] \end{aligned} \tag{6.52}$$

where

$$I_{k,m}(t-\Delta t) = -i_{k,m}(t-\Delta t) - \frac{2C}{\Delta t}[v_k(t-\Delta t) - v_m(t-\Delta t)] \tag{6.53}$$

This can be visualised as shown in fig. 6.6c(ii).

6.5.2 Transient solution

As described in section 4.3, the EMTP simulation method refers to, firstly, reducing the whole transmission network to its components, that is, combinations

of equivalent node voltage source, branch current source, transmission line, resistance, inductance and capacitance. Secondly, after nodal analysis has been applied to the network, a set of equations is set up which simulate the behaviour of each component, e.g., using the fundamental building block relationships as derived in the previous section and the transmission line equations as derived in section 4.4. Finally, the whole set of equations must be solved numerically in the time-domain step by step and each new output sample is derived from a few already known past history terms. In terms of matrix algebra, the following matrix equations can be written for a system of n nodes [6].

$$[\mathbf{G}]\,[\mathbf{V}(t)] = [\mathbf{I}(t)] - [\mathbf{I}_{hist}] \tag{6.54}$$

where

$$\begin{array}{rcl} [\mathbf{G}] & = & n \times n \text{ symmetric nodal conductance matrix,} \\ [\mathbf{V}(t)] & = & n \times 1 \text{ column vector for nodal voltages,} \\ [\mathbf{I}(t)] & = & n \times 1 \text{ column vector for current sources, and} \\ [\mathbf{I}_{hist}] & = & n \times 1 \text{ column vector for known history terms.} \end{array}$$

If some elements in the voltage vector are known, equation 6.54 can be partitioned into a set of unknown voltages, $[\mathbf{V}_a(t)]$, and a set of known voltages, $[\mathbf{V}_b(t)]$.

$$\begin{bmatrix} \mathbf{G}_{aa} & \mathbf{G}_{ab} \\ \mathbf{G}_{ba} & \mathbf{G}_{bb} \end{bmatrix} \begin{bmatrix} \mathbf{V}_a(t) \\ \mathbf{V}_b(t) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_a(t) \\ \mathbf{I}_b(t) \end{bmatrix} - \begin{bmatrix} \mathbf{I}_{a_{hist}} \\ \mathbf{I}_{b_{hist}} \end{bmatrix} \tag{6.55}$$

The unknown voltages, $[\mathbf{V}_a(t)]$ are then be found by solving the following equations:

$$[\mathbf{G}_{aa}]\,[\mathbf{V}_a(t)] = [\mathbf{I}_a(t)] - [\mathbf{I}_{a_{hist}}] - [\mathbf{G}_{ab}]\,[\mathbf{V}_b(t)] \tag{6.56}$$

The actual computation in the EMTP proceeds as follows: Matrices $[\mathbf{G}_{aa}]$ and $[\mathbf{G}_{ab}]$ are built, and $[\mathbf{G}_{aa}]$ is triangularized with ordered elimination and exploitation of sparsity. In each time step, the vector on the right-hand side of equation 6.56 is 'assembled' from known history terms, and known current and voltage sources. Then the system of linear equations is solved for $[\mathbf{V}_a(t)]$. Before

proceeding to the next time step, the history terms are then updated for future use.

6.5.3 Steady state solution

Similarly, for steady state solution of an electrical circuit, the following basic relationships can be used to simulate the components R, L and C.

$$I_{k,m} = \frac{1}{R}[V_k - V_m] \tag{6.57}$$

$$I_{k,m} = \frac{1}{j\omega L}[V_k - V_m] \tag{6.58}$$

$$I_{k,m} = j\omega C[V_k - V_m] \tag{6.59}$$

where $\omega = 2\pi f$ and f is the steady state power frequency.

Using the above relationships, together with the steady state transmission line equations and following similar procedure as described in the previous section, the following matrix equations can be formed for a system of n nodes.

$$[\mathbf{Y}][\mathbf{V}] = [\mathbf{I}] \tag{6.60}$$

where

$[\mathbf{Y}]$ = $n \times n$ symmetric nodal admittance matrix,
$[\mathbf{V}]$ = $n \times 1$ column vector for nodal voltages (complex phasor values) and
$[\mathbf{I}]$ = $n \times 1$ column vector for current sources (complex phasor values).

Again, equation 6.60 can be partitioned into a set of unknown voltages, $[\mathbf{V}_a]$, and a set of known voltages, $[\mathbf{V}_b]$. The unknown voltages $[\mathbf{V}_a]$ can then be found by solving the following equations:

$$[\mathbf{Y}_{aa}][\mathbf{V}_a] = [\mathbf{I}_a] - [\mathbf{Y}_{ab}][\mathbf{V}_b] \tag{6.61}$$

6.5.4 Fault transient modelling

The fault sequences modelled are the same as described in previous sections for the frequency domain technique and the time-convolutional technique. A three phase double end fed transmission network is shown in fig. 4.2. With this particular model any type of earth fault, fault break-off or release can be treated by simulating the closure or opening of the appropriate switches at the fault point. Opening, and subsequent reclosure, of the circuit breakers is likewise simulated by the operation of the appropriate switches, according to the sequence associated with any type of autoreclosure cycle under consideration.

It should be emphasised that a frequency dependent line parameter model is used to simulate the electromagnetic transients where the high frequency components are dominant, and throughout all the sequences of the electromagnetic transient simulation, the generator is represented by a constant lumped series subtransient reactance and the generator speed is assumed constant. Whilst this assumption is justified for the relatively short time period of prefault clearance, it is not strictly valid in the so-called 'dead period' once the fault has been broken off. Hence, it is necessary to incorporate an actual generator model into the simulator in order to be able to produce the low frequency effects associated with the electromechanical transients.

6.5.5 Electromechanical transient modelling

The schematic diagram of a single machine power system is shown in fig. 5.1. A full turbo-alternator model with governor turbine and AVR control and a step up generator transformer, is connected to through a transmission network to an

ideal receiving end source. The transformer used is delta-star connected with star side neutral grounded. Frequency dependent transmission line parameters are used to simulate the transmission line. An IEEE type-1 AVR as shown in fig. 5.4 is used to implement the excitation control. An IEEE steam governor and valve control system as shown in fig. 5.5 is used for governor turbine control. Detailed descriptions of each component of fig. 5.1 for a single machine power system can be found in Chapter 5.

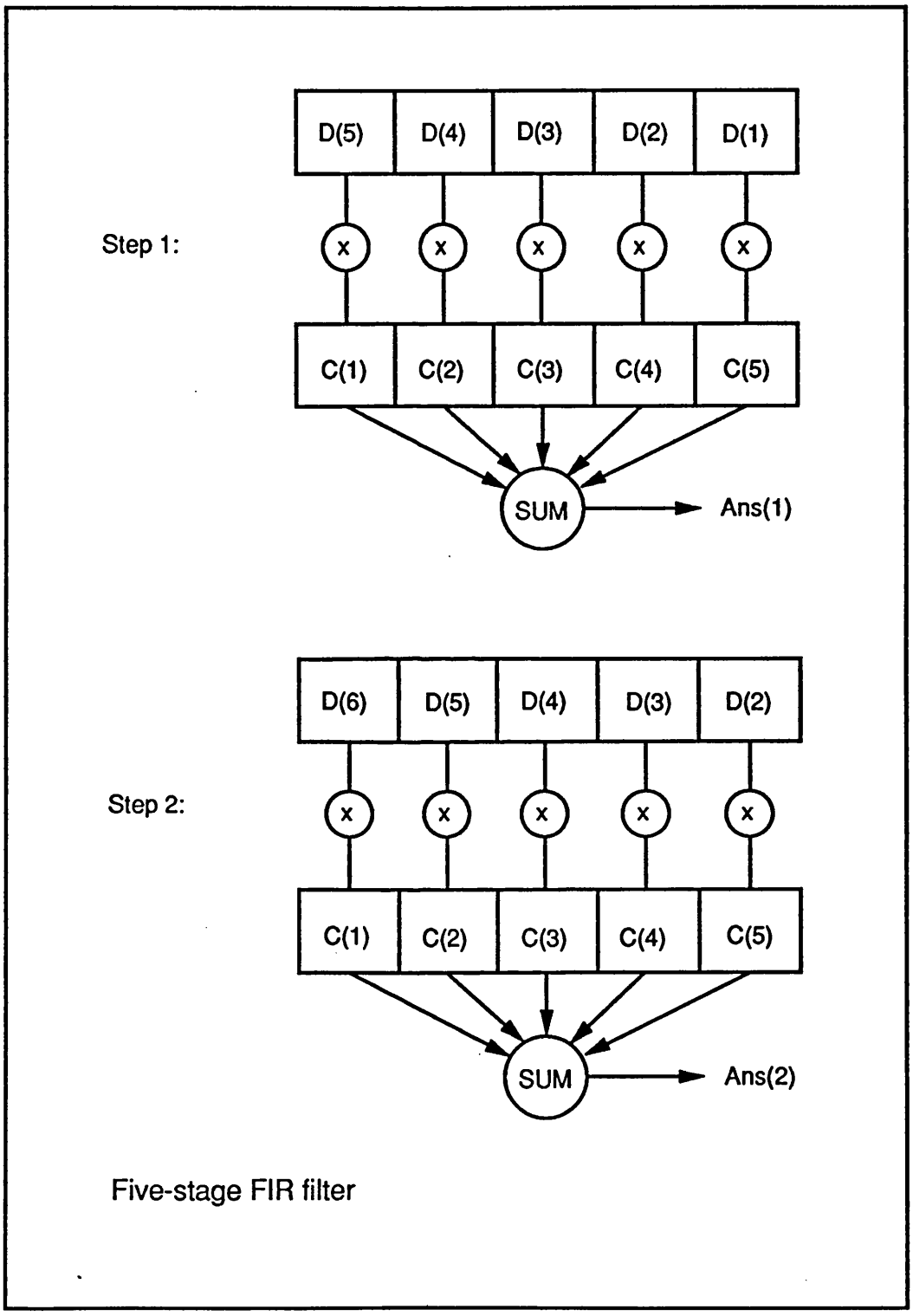

Figure 6.1: Numerical approximations for time convolution integral

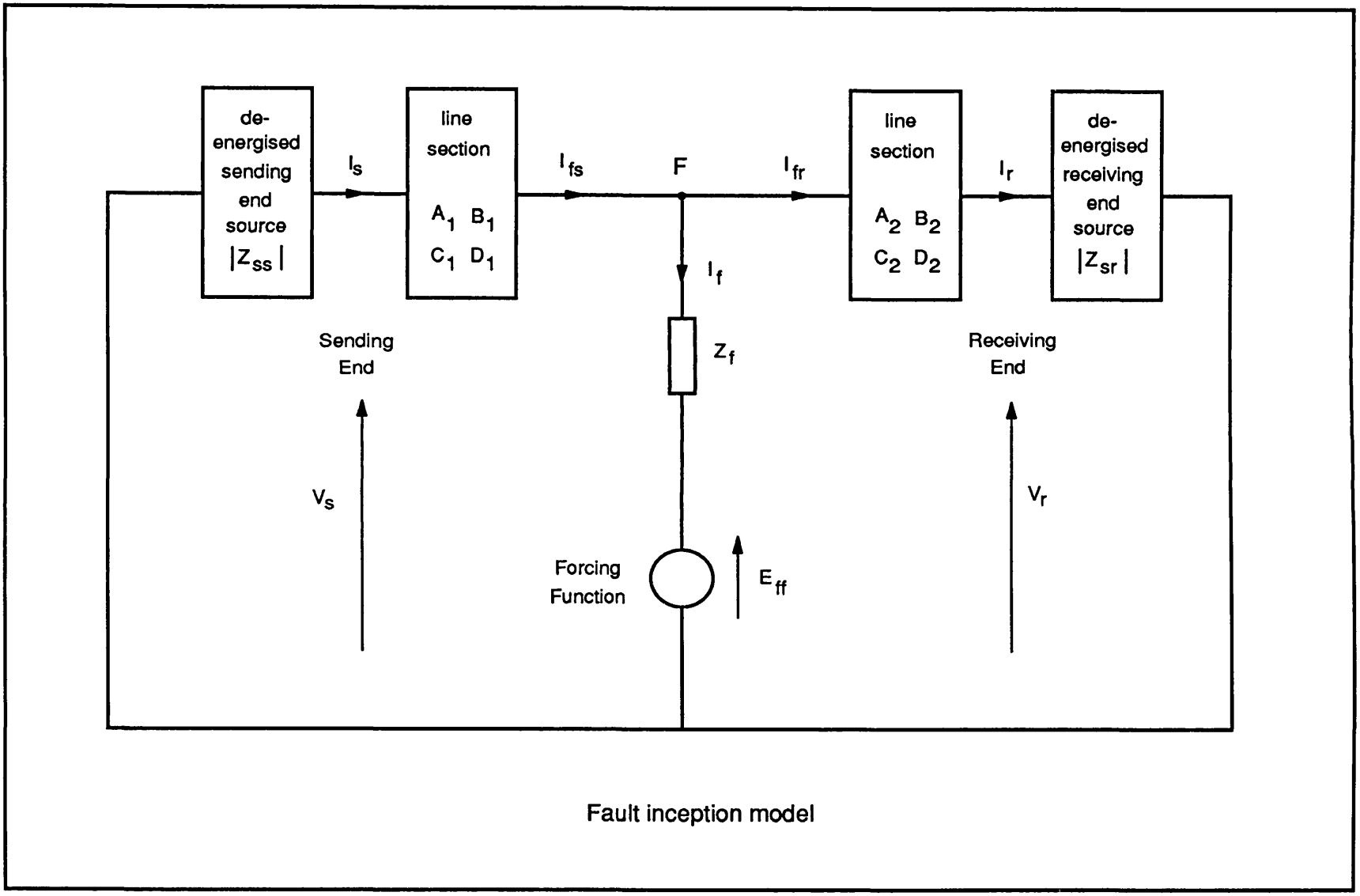

Figure 6.2: Fault inception model

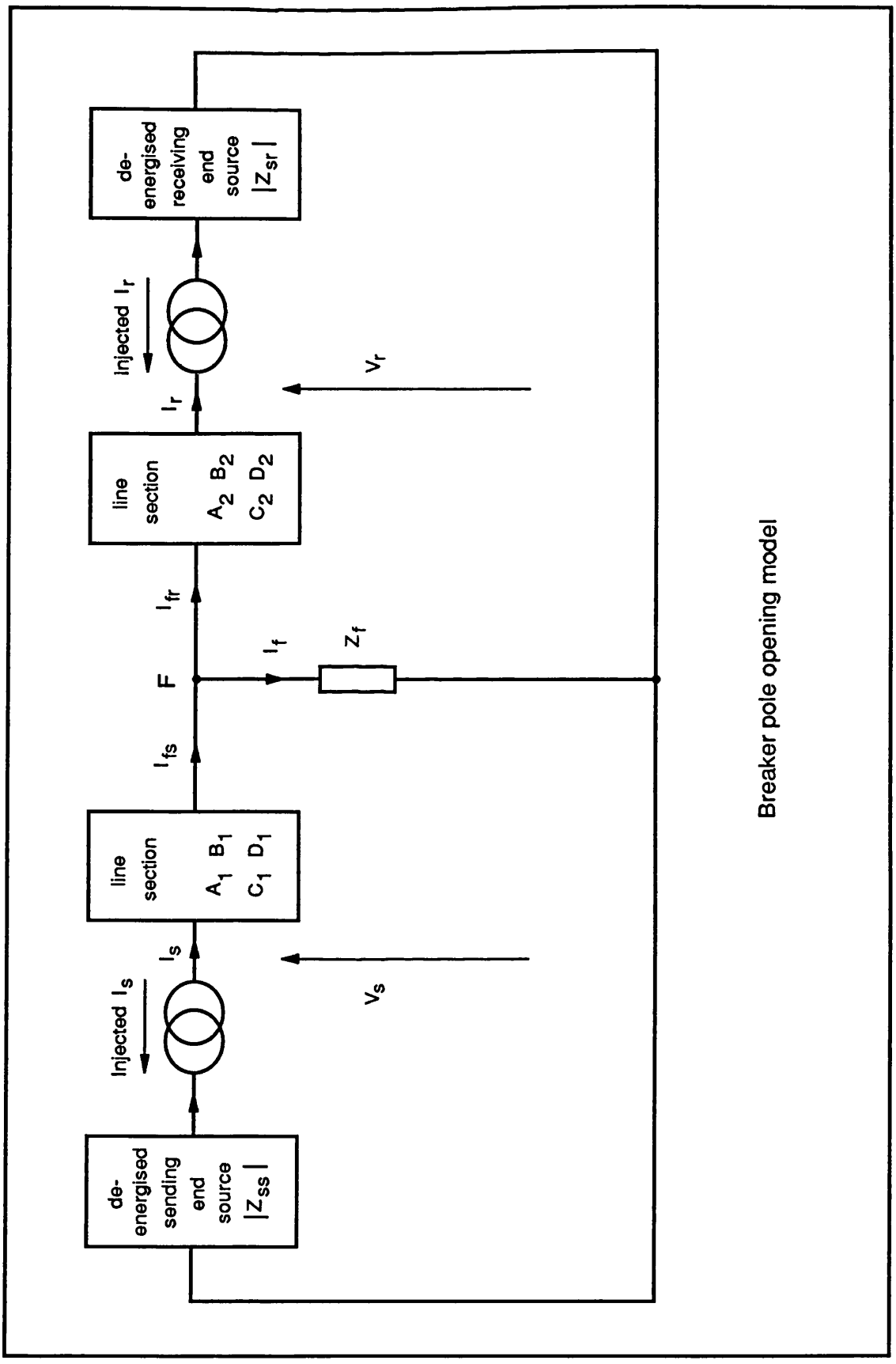

Figure 6.3: Circuit breaker pole opening model

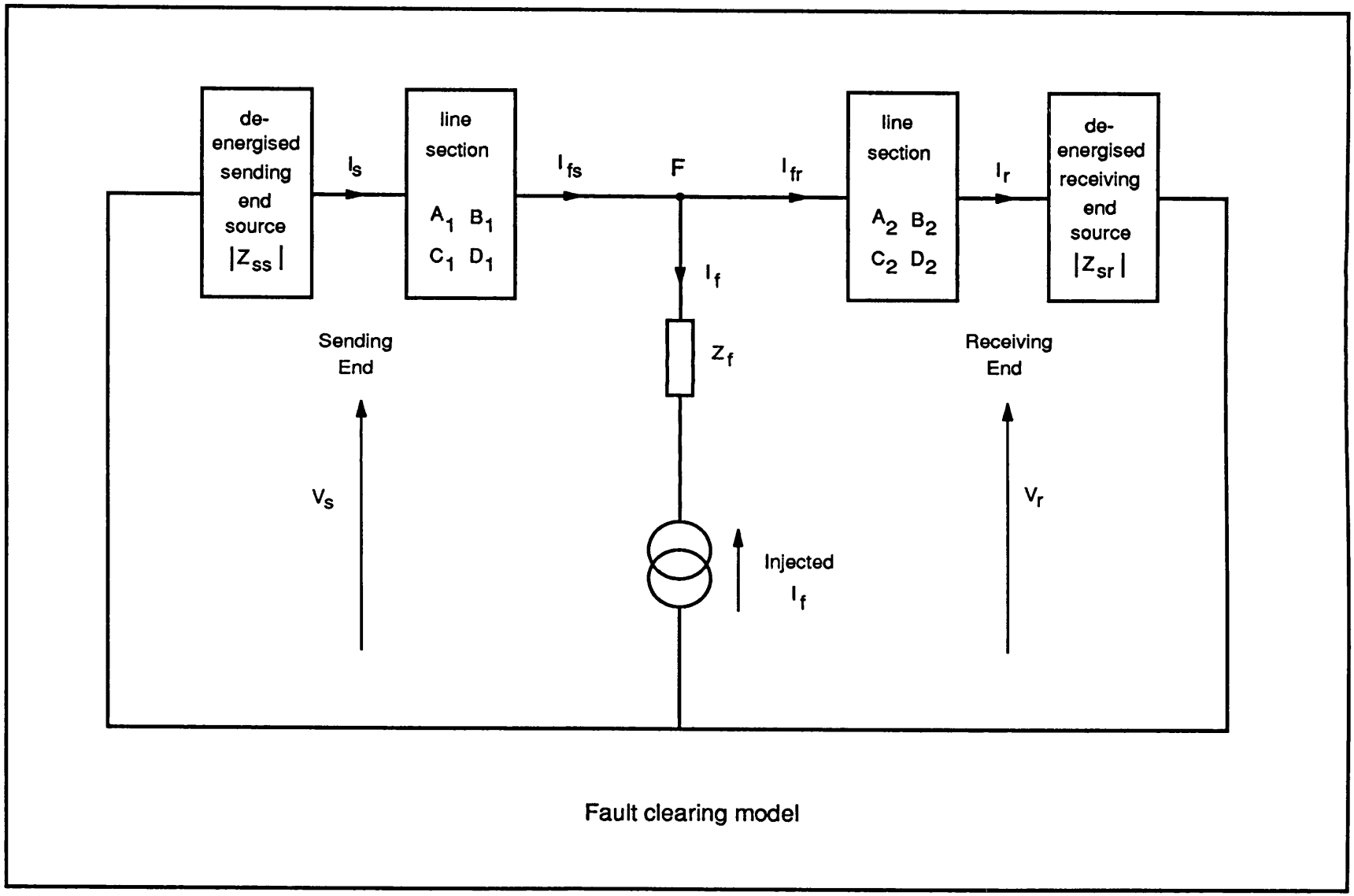

Figure 6.4: Fault clearing model

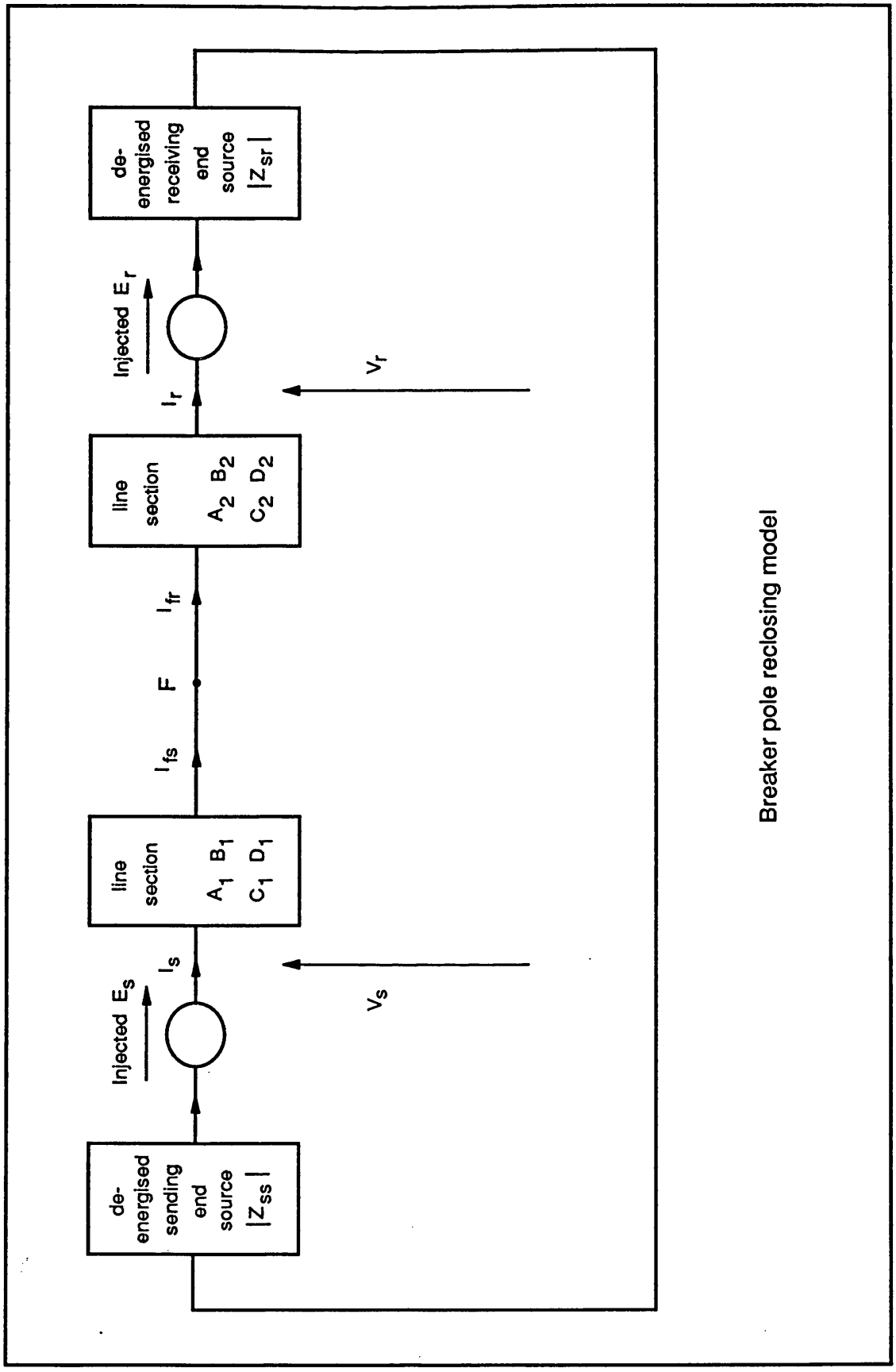

Figure 6.5: Circuit breaker pole reclosing model

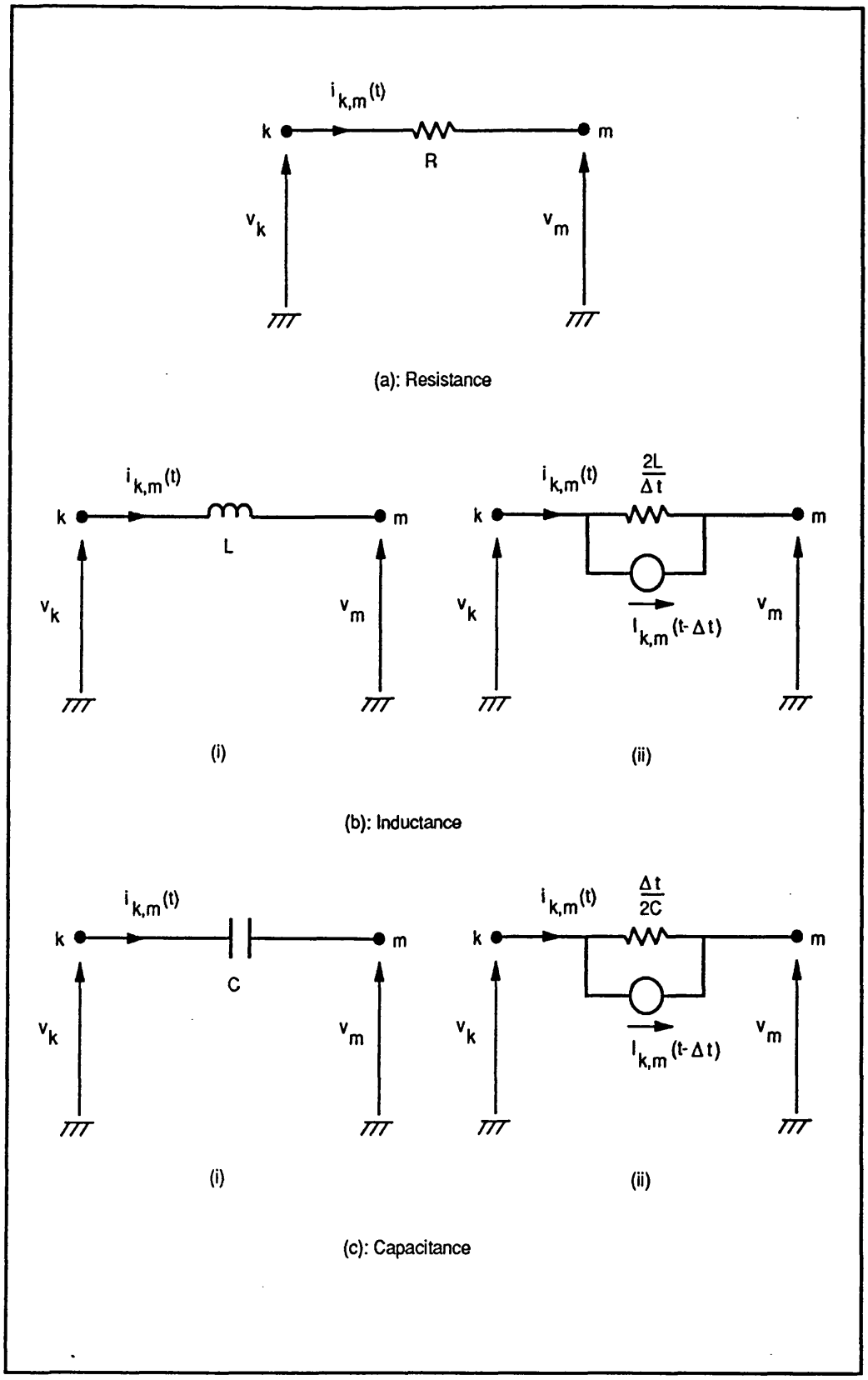

Figure 6.6: Simulation of basic power system components

Chapter 7

Simulation hardware and software

7.1 Introduction

This chapter describes the hardware and software used for the project. Optimizations have been carried out to the simulating codes in order to achieve the best computational time. Parallel computation algorithms, which make use of the Helios multiprocessing techniques [40] running on T800 transputers [21], have been designed for the simulating codes. In order to attain peak performance, handcoding has been made to the simulation codes using the Intel i860 microprocessor's parallel processing capabilities [23].

7.2 Hardware

The advent of large digital computers in the 1960s paved the way for unprecedented developments in power system analysis and with them the availability of

a more reliable and economic supply of electrical energy with tighter control of the system frequency and voltage levels. Probably the main development of the decade in power system analysis has been the change of emphasis from mainframe-based to interactive analysis software. Until IBM introduced the PC/AT in 1984, it was out of the question to use a PC to perform power system analyses.

The 32-bit architecture and speed of the Intel-80386 and 80486 microprocessor chips combined with the highly increased storage capability and speed of hard disks has made it possible to perform the simulation studies on the PC. Moreover, some compilers such as the Salford University's 386 Fortran compilers have become available which are capable of handling virtual memory management and code requirements for running very large simulator programs on the PC.

7.2.1 Hardware used in the project

The simulator software was originally developed under the Helios Operating System [20], which was hosted in an Intel 80386 [22] based IBM compatible PC with several T800 transputer [21] plug-in boards. In order to obtain benchmark calculation time for the simulator, the programs were then ported to an Intel 80386 based PC with an Intel i860 [23] plug-in board. The EMTP programs were developed on an Intel 80386 based IBM compatible PC with an Intel 80387 maths-coprocessor and 8-Mbytes memory.

Intel 80386: The 80386 [22] is a powerful full 32-bit microprocessor which has internal 32-bit registers with 32-bit buses for both data and address. It supports a virtual memory system and has other advanced features which make it faster in operation and offer more flexible memory management

than an Intel-80286 microprocessor [22]. It can operates in two modes. The first, the real mode, forces the 80386 to behave and respond exactly like the Intel-8086 microprocessor. The second, the protected mode, reserves a pre-determined amount of memory for program execution; this memory is protected from use by any other program. Multi-tasking capabilities originate in protected mode. By using the protected mode, several programs can run concurrently without affecting one another. A maths-coprocessor 80387 can be used to speed up the floating point calculations.

Inmos T800: The IMS T800 transputer [21] is a 32-bit CMOS microcomputer with a 64-bit floating point unit. It has 4-Kbytes of on-chip RAM for high speed processing, a configurable memory interface and four standard INMOS communication links. It provides high performance floating point and arithmetic operations. Fig. 7.1 shows the block diagram of an IMS-T800. The transputer's real power and flexibility is its ability to be linked to other transputers by high bandwidth serial links. The move from single processor operation to multi-processing is simple and software can be reconfigured for multiple transputer systems without even having to recompile the source. This means that small systems can be used to minimise development costs while the full power of multiprocessing can be exploited once the application has been proven.

Intel i860: The Intel *i*860 microprocessor [23] delivers supercomputing performance in a single VLSI component. The 32/64 bit architecture of the *i*860 balances integer, floating point, and graphics performance. Its parallel architecture achieves high throughput with RISC design techniques, multiprocessor support, pipelined processing units, wide data paths, large on-chip caches, 2.5 million transistor design, and fast 0.8-micron silicon technology. Fig. 7.2 illustrate the registers and data paths of an *i*860 mi-

croprocessor. According to the data given by S.S. Fried [24], amongst all the other microprocessors, the Intel i860 can achieve the highest MFLOPS operations at the time of publication.

7.3 Software overview

1. **Calculation overview:**

   Chapter 4 describes the general formulations for modelling the fault transients of a single conductor line and a three phase line. The system equations, which relate the sending end, receiving end and the fault point parameters, expressed in the frequency domain are:

$$\begin{bmatrix} I_f \\ I_s \\ I_r \end{bmatrix} = \begin{bmatrix} Y_A & Y_B & Y_C \\ Y_D & Y_E & Y_F \\ Y_G & Y_H & Y_I \end{bmatrix} \begin{bmatrix} E_{ff} \\ E_s \\ E_r \end{bmatrix} \tag{7.1}$$

$$\begin{bmatrix} E_{ff} \\ E_s \\ E_r \end{bmatrix} = \begin{bmatrix} Z_A & Z_B & Z_C \\ Z_D & Z_E & Z_F \\ Z_G & Z_H & Z_I \end{bmatrix} \begin{bmatrix} I_f \\ I_s \\ I_r \end{bmatrix} \tag{7.2}$$

   If the transients are calculated in the frequency domain, then the above system equations can be solved directly. For example,

$$I_f = Y_A E_{ff} + Y_B E_s + Y_C E_r \tag{7.3}$$

   If the time domain convolutional technique is used to calculate the transients, then the following system equations must be solved using convolutions.

$$\begin{bmatrix} i_f(t) \\ i_s(t) \\ i_r(t) \end{bmatrix} = \begin{bmatrix} y_a & y_b & y_c \\ y_d & y_e & y_f \\ y_g & y_h & y_i \end{bmatrix} * \begin{bmatrix} e_{ff}(t) \\ e_s(t) \\ e_r(t) \end{bmatrix} \tag{7.4}$$

$$\begin{bmatrix} e_{ff}(t) \\ e_s(t) \\ e_r(t) \end{bmatrix} = \begin{bmatrix} z_a & z_b & z_c \\ z_d & z_e & z_f \\ z_g & z_h & z_i \end{bmatrix} * \begin{bmatrix} i_f(t) \\ i_s(t) \\ i_r(t) \end{bmatrix} \tag{7.5}$$

   For example,

$$i_f = y_a * e_{ff} + y_b * e_s + y_c * e_r \tag{7.6}$$

2. **Fault sequence overview:**

   The fault sequence of an autoreclosure operation can be summarized as shown below:

   Event 0 : Pre-fault steady state conditions.

   Event 1 : A fault to ground occurs at the fault point on the line at time t_{fault}.

   Event 2 : The circuit breaker poles of both the receiving end and the sending end open simultaneously at time t_{open}.

   Event 3 : The fault is cleared at time t_{clear}.

   Event 4 : The circuit breaker poles of both the receiving end and the sending end reclose simultaneously at time $t_{reclose}$.

   It is assumed in the simulation that the circuit breaker poles are opened at time t_{open} and reclosed at time $t_{reclose}$ *simultaneously* respectively.

3. **Dynamic storage management overview:**

   Dynamic memory allocation facilities are utilized by the simulator. A simple form of heap memory management is provided. It allows the simulator to repeatedly request allocation of a 'fresh' region of memory and perhaps later to deallocate such a region when it is no longer needed. Explicitly de-allocated regions are recycled by the storage manager for satisfaction of further allocation requests.

4. **Calculation library overview:**

   Since C does not have data types such as complex number, complex number sparse matrix and complex number vector, a complex number calculation library, has been written for the simulator. Appendix D shows the functions which are defined in the complex number library.

5. **Parallel programming overview:**

   Helios [20] is an operating system that is designed to run on a wide range of transputer based architecture. Helios supports certain environments where transputers simply act as accelerators for existing machines such as IBM-PC/AT and SUN workstations. Here the host processor retains many of the input/output functions of the stand-alone machine. A standard server for the host machine, provided with the system software, acts as an interface to the transputer back end. This server allows the user to access all the standard features of the host environment, and appears to the Helios network like any other Helios processor.

   CDL [40] stands for Component Distribution Language. It enables parallel programming to be carried out under the Helios Operating System. The purpose of CDL is to provide a high-level approach to parallel programming, where the programmer defines the program components and their relative interconnections and allows Helios to take care of the actual distribution of these components over the available physical resources.

7.4 Parallel processing algorithm

The simulator can mainly be divided into a **master task** and a number of **worker tasks**. Based on the farming techniques of the Helios CDL, the simulator algorithm was written in such a way that the transient calculations could be done by efficient job divisions among a number of specified workers. The number of workers, which can theoretically be any integer values, can be specified by the user. However, a sensible choice is that the number of workers should be chosen such that the overall calculation time is minimized .

Assuming that the number of workers specified by the user is 3, the simulator tasks can then be represented as shown in fig. 7.3. The master task mainly does three main tasks. Firstly, it reads in study database from the study files, where the system Y matrix and Z matrix, which are defined in Chapter 4.2, as well as the fault forcing function E_{ff} are stored. It then initializes the workers and starts the fault autoreclosure sequence by sending and receiving data from the workers. Finally, after finishing all the events, it then writes the resultant waveforms of the sending end, receiving end and the fault point voltages and currents to different files, which can then be plotted or used for testing relays.

7.4.1 Read study database

Due to the travelling wave characteristics, it is a necessary to evaluate the system equations over the entire frequency spectrum of the fault disturbance in order to obtain the complete transient response. For real time simulation purposes, the frequency spectra of the system Y-matrix, Z-matrix and the forcing fault point voltage function, E_{ff} are calculated off-line and stored in several data files. This data is read in by the master after the program has been started.

7.4.2 Initialization of workers, communications and transient calculations

After reading in all the database, the master then initializes all the workers so that the workers can allocate suitable memory and be ready for receiving data and carry out transient calculations. When starting up the task force, the task force manager will automatically create various pipes and standard streams, which are used for the communications between different tasks. Since the communication

overhead cost affects the efficiency of the multiprocessing algorithm quite significantly, the algorithm has been designed so that the master task could write to several workers or read from several workers at the same time. This is achieved

Master Streams	POSIX No.	Function
stdin	0	Input from user
stdout	1	Display text to user
stderr	2	Report errors to user
auxin1	4	Data from first worker
auxout1	5	Data to first worker
auxin2	6	Data from second worker
auxout2	7	Data to second worker
auxin3	8	Data from third worker
auxout3	9	Data to third worker
auxin4	10	Data from fourth worker
auxout4	11	Data to fourth worker

Table 7.1: Master stream number assignments for 4 workers

Slave Streams	POSIX No.	Function
stdin	0	Data from master
stdout	1	Data to master
auxin1	4	Data from another worker
auxout1	5	Data to another worker

Table 7.2: Worker stream number assignments

by having multiple threads within the master task. The master task consists of having a single main thread to interact with the user and display, and a number of additional threads to deal with the workers. Table 7.1 and 7.2 show the streams used for communications between different tasks. Data are transferred between different tasks by using **read** and **write** routines. Suppose the master needs to write $50bytes$ data to worker 2, auxout2 connects to this worker, so one way of doing it is:

$$result = write(7, buffer, 50)$$

The slave can read this data by reading from its standard input:

$$result = read(0, buffer, 50)$$

Notice that the master and the worker agree exactly about the amount of data to be transferred. Also, the data transfer is synchronised: if the master tries to write before the worker is ready, it gets suspended until the master has written the data or until the master closes the stream, e.g., when it exits.

Assuming all the workers have been successfully initialized, the master can then transfer the relevant data to the workers. At this point, the workers are ready to do the requested calculations upon the requests of the master. When all the workers have finished the requested work, the master then receives the resultant data from the workers and the cycle is then repeated until all the sequence of events have been simulated.

7.4.3 Summary of the multiprocessing algorithm

The multiprocessing algorithm of the electromagnetic transient simulator can be summarised as shown below:

- **Master**
  - **Main Thread**
    1. Determine the number of workers.
    2. Allocate memory for study database.
    3. Read in study database.
    4. Initialise all the workers.

5. Start new threads, one thread per worker, and wait for all the workers to be ready.
6. For the fault sequence of events to be simulated, switch into one of the following routines:
   (a) Calculation of steady state values.
   (b) Modify the forcing functions as required.
   (c) Update the data for transient calculations as required.
   (d) Signal all the workers to do one of the following transient calculations:
      i. Fault inception.
      ii. Breaker pole opening.
      iii. Fault clearance.
      iv. Breaker pole reclosure.

– **For every slave, in a separate thread**

1. Initialise a worker with the correct parameters.
2. Check that the worker can allocate enough memory.
3. Send the appropriate part of the system data to the worker for the transient calculations.
4. Signal that this worker is ready.
5. Forever
   (a) Send the current job identity to the worker.
   (b) Send the updated data to the worker.
   (c) Read resulting vectors back from the worker.
   (d) Signal that the worker has finished its job.

• **Worker**

1. Read the correct parameters.

2. Allocate enough memory for this worker's part of transient calculations.

3. Forever

   (a) Read identity of the current job to be performed.

   (b) Read updated data for a particular transient calculations.

   (c) Do the transient calculations.

   (d) Send back the results to the master if necessary, and at the same time send results, which are requested by another worker, directly to that specified worker.

7.5 Optimizations for frequency technique

In order to achieve the 'real time' requirement for the opening of the circuit breakers, the simulating codes have been modified and improved. The following methods have been found to be most suitable for optimizing the electromagnetic transient simulation:

1. One processor for one or several variables to be calculated, say, I_{sa}, I_{sb}, V_{sa}, V_{sb}, etc.

2. Precalculate all the parameters which could be done before fault inception.

3. Optimizations of the forward and backward FFTs.

4. Precalculations of the inversions of all the submatrices.

7.5.1 Simulation stages

Prefault stage

1. Calculations of the spectra of universal system matrices.
2. Calculations of the spectra of fault forcing functions.
3. Pre-calculations of the spectra of all the variables which will be used after fault inception.
4. Calculations of all the prefault steady state values.

Fault inception stage

1. Calculations of the fault inception transients in the frequency domain.
2. Transforming the results from the frequency domain back to the time domain.

Circuit breaker pole opening stage

1. Transformation of six currents, namely, I_{sa}, I_{sb}, I_{sc}, I_{ra}, I_{rb} and I_{rc} in case of three phase autoreclosure from real vectors to complex vectors in the time domain. This involves the numerical solution of the modified Fourier transform as given in eqn. 6.42.

   Optimizations used :

The following methods are used to speed up the numerical evaluation of eqn. 6.42:

(a) Precalculate the spectrum of $e^{-j\frac{n\pi}{N_2}}$, where $N_2 = 2\times$ simulated samples, and $n = 0, 1, \cdots, N_2 - 1$.

(b) Precalculate the spectrum of $e^{-at} \times \Delta t/2$;

2. Transformation of the six currents from the time domain to frequency domain by using forward FFTs.

Optimizations used :

Detailed numerical technique for solving the FFTs is given in Appendix C.

(a) Butterfly algorithm was adopted. The resulting code is about three to four times faster than the original Fortran code.

(b) Precalculations of the twiddle factors. The resulting code is about 15 per cent faster.

(c) A combination of DIT (Decimation in time) for backward FFT and DIF (Decimation in frequency) for forward FFT to eliminate bit-reverse routines was tried in the simulator. The aim was to try to use the scrambled FFT outputs in the frequency domain. However, it was proved that the scrambled outputs could not be used directly in the simulator. The reason for this is because that the number of samples in forward and backward FFTs were different.

(d) Handcoding of FFTs according to i860 architecture. Dual-instruction and dual-operation modes of the i860 were used. Two butterflies (odd and even) per inner-loop were carried out. The resulting code is about two to three times faster.

(e) Precalculations of the swapping vectors.

(f) Precalculations of $e^{-j\frac{n\pi}{N_2}}$ and $e^{-at} \times \Delta t/2$ which are required for transforming a floating point vector to a complex vector (see eqn. 6.42) before using the forward FFT.

(g) Single precision instead of double precision.

3. Performing transient calculations in frequency domain. This involves :

(a) Selecting appropriate submatrices from the universal system matrices and solving the selected equations for the appropriate parameters. For example,

$$\begin{bmatrix} I_s \\ I_r \end{bmatrix} = \begin{bmatrix} Y_E & Y_F \\ Y_H & Y_I \end{bmatrix} \begin{bmatrix} E_s \\ E_r \end{bmatrix} \tag{7.7}$$

(b) $V_{sa} = E_{sa} - Z_{ss} \times I_{s(a,b,c)}$

(c) $V_{sa} = V_{sa} \times \sigma \times \Delta\omega$

Optimizations used:

(a) Precalculate spectra of system matrices.

(b) Precalculate spectra of the inversions of the sub-matrices.

(c) Precalculate the Z_{ss} spectra.

(d) Precalculate the $\sigma \times \Delta\omega$ spectra.

4. Transformation of the results from the frequency domain back to the time domain by using the backward FFTs. Similar optimizations as stated in forward FFT were also carried out for the backward FFT.

5. Transformation of the resulting complex vectors obtained from the backward FFTs to real vectors.

Optimizations used :

(a) Precalculate the E_b and E_a spectra, where

$$E_a = \frac{e^{atn}}{\pi}$$

$$E_b = e^{j\pi n/N}$$

for N = number of simulated samples and $n = 0, 1, \cdots, N-1$.

7.5.2 Handcoding using i860 architecture

In order to achieve the best performance from the i860, the transistors in the CPU must be kept in constant and productive use. But this does not happen automatically. In scalar mode, the i860 does not do much better than the Intel-80486/80487 microprocessors. To attain peak performance, the processor's memory systems, the chip's pipelining and parallel-processing capabilities must be exploited. This requires handcrafting and fine-tuning the simulation codes according to the i860's architecture.

As shown in fig. 7.2, each i860 processor has 32 integer registers and 32 floating-point registers, each 32 bits wide. Instructions feed out of a 4K-byte, 64-bit-wide cache that can drive both the RISC core and the FPU simultaneously through independent 32-bit instruction buses. This is one way of parallelism with the i860. Data feeds out of an 8K-byte, 128-bit-wide cache that can drive two real arguments at a time at the adder, multiplier, or graphics unit. For single precision, the adder is a three-stage pipeline, as is the multiplier, and these two units can hook together in a variety of ways. This is another form of parallelism.

Floating-point operations

The architecture of the floating-point unit uses parallelism to increase the rate at which operations can be started. The pipelined architecture treats each operation as a series of more primitive operations, called stages, that execute in parallel.

The i860 has four pipelines, namely, the external memory loader, the adder, the multiplier, and the graphics unit. The adder and multiplier can load from registers, the data cache, or external memory. The ability to pipeline loads from external memory is crucial for vector operations on large matrices. The number of pipeline stages ranges from one to three. A pipelined instruction with a three-stage pipeline writes to its *fdest* (fig. 7.4) the result of the third prior instruction. Similarly, a pipelined operation with a one-stage pipeline stores the result of the prior operation.

In addition to the pipelined execution mode described above, the i860 can also execute floating point in scalar mode. In this mode, the unit does not start a new operation until the previous operation is completed. The scalar operation passes through all stages of its pipeline before a new operation is introduced, and the result is stored automatically. A scalar mode is used when the next operation depends on results from the previous few floating-point operations.

Dual-operation and dual-instruction mode

In the *dual-operation mode*, the adder and multiplier work in concert. There are 62 ways to chain together the i860's adder and multiplier. Fig. 7.4 shows the full set of possibilities surrounding the adder and multiplier. Three special registers *KR, KI, and T*, can be used to store values from one dual-operation instruction and supply them as inputs to subsequent dual-operation instructions. Fig. 7.5 illustrates the actual data path for the multiply-accumulate instruction, one of the dual-operation instructions. As shown in fig. 7.5, the multiplier feeds the adder, while the adder's results recirculate back through the adder.

In the *dual-instruction mode*, the RISC core loads floating-point registers while the FPU runs in parallel. This mode is entered by using the dual-instruction mode transitions as shown in fig. 7.6. When executing in the dual-instruction mode, the instruction sequence consists of 64-bit aligned instruction pairs with a floating-point instruction in the lower 32-bits and a core instruction in the upper 32-bits. Thus, two instructions can be fetched at the same time from the instruction cache and pipelined loads and pipelined computations can be carried out at the same time.

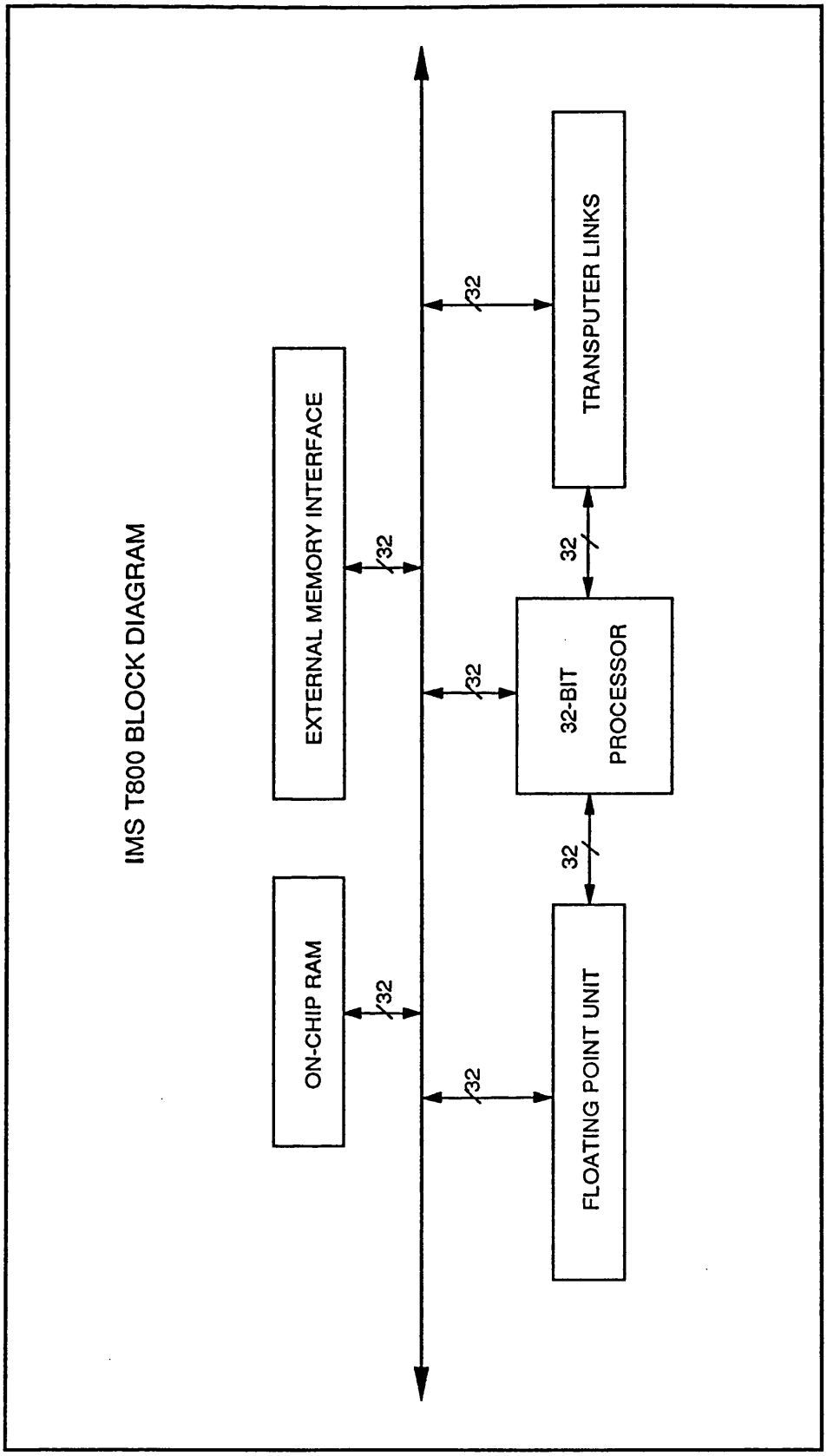

Figure 7.1: T800 transputer block diagram.

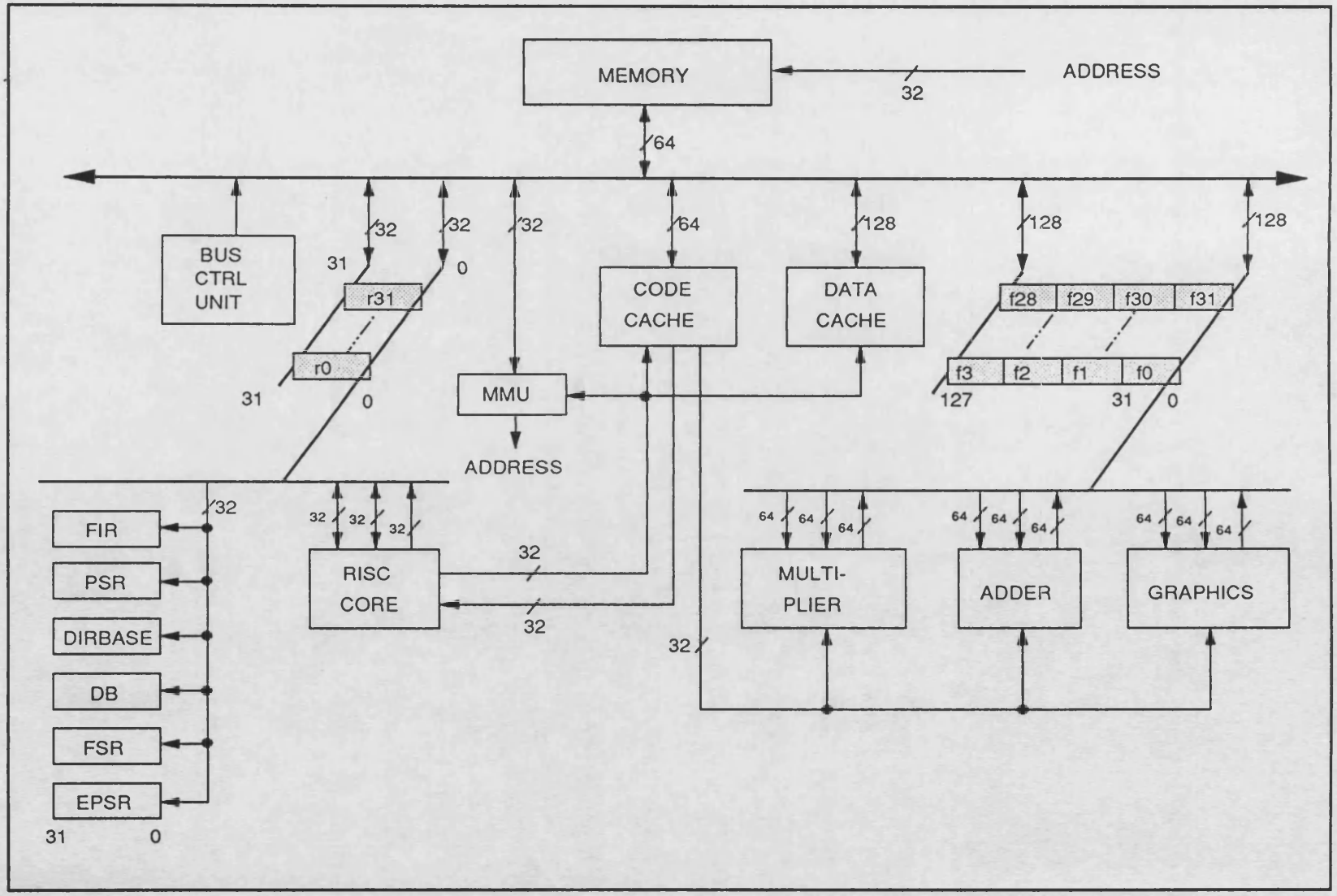

Figure 7.2: i860 registers and data paths.

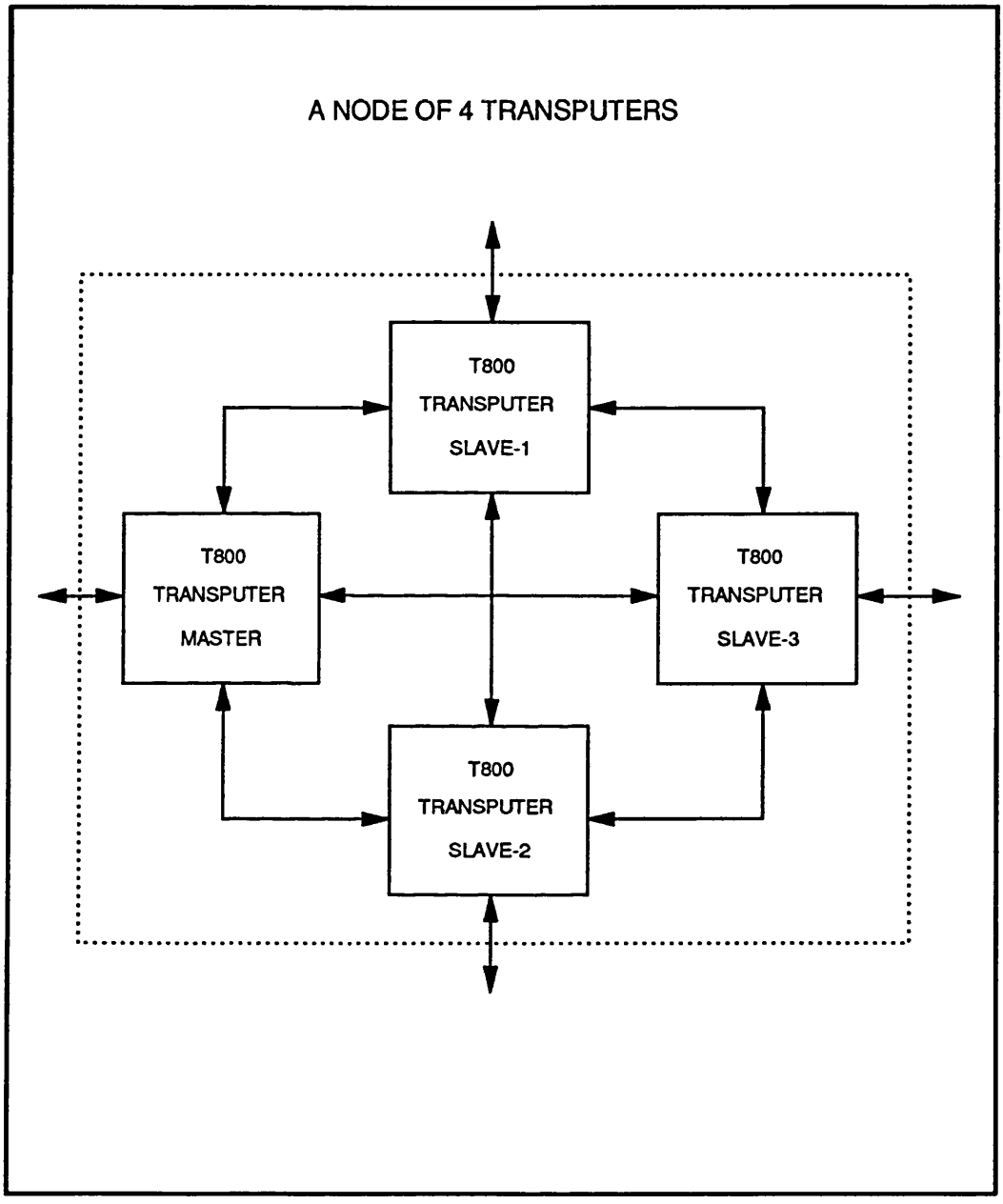

Figure 7.3: Simulation tasks

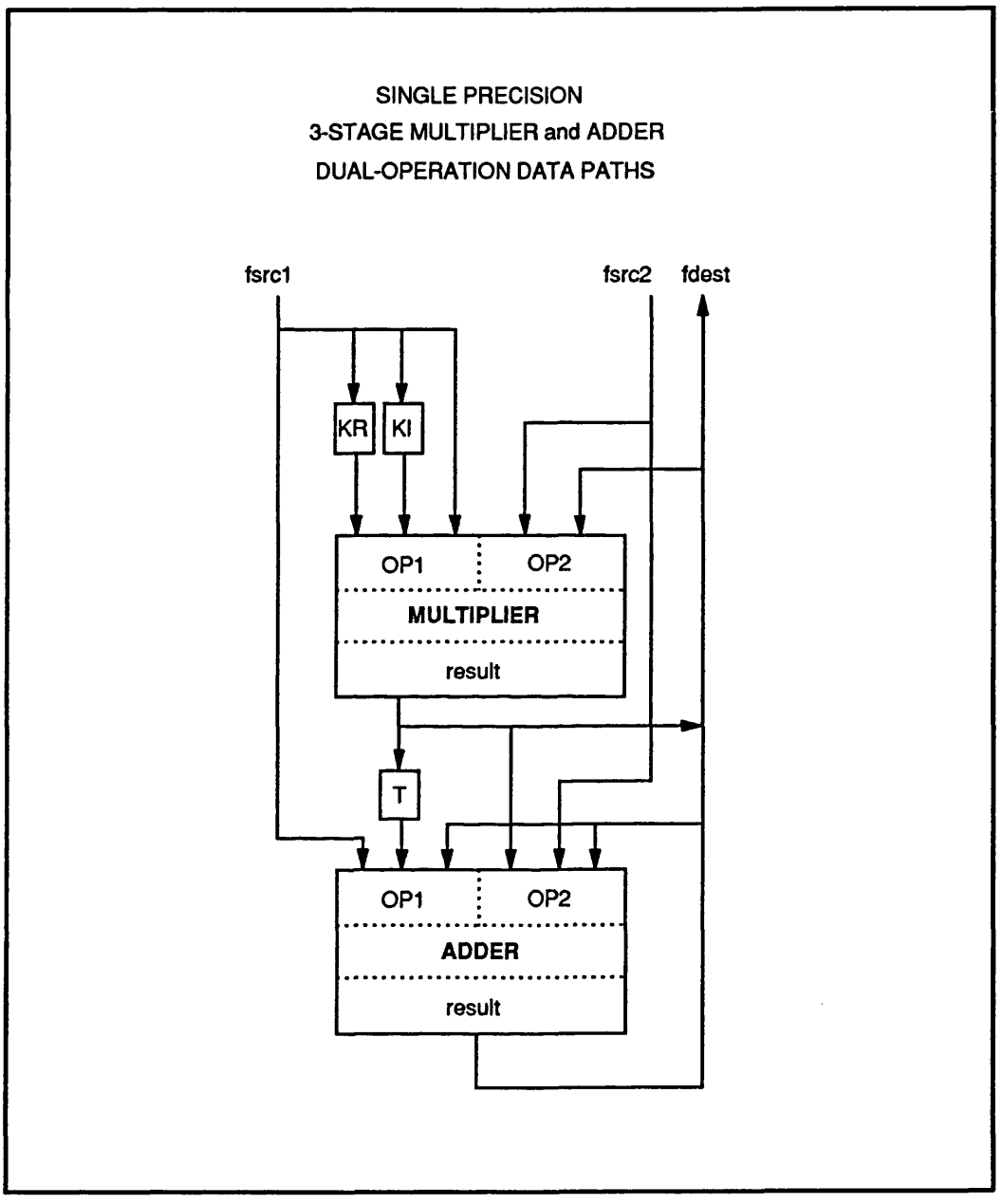

Figure 7.4: Full set of data paths for chaining the adder and the multiplier.

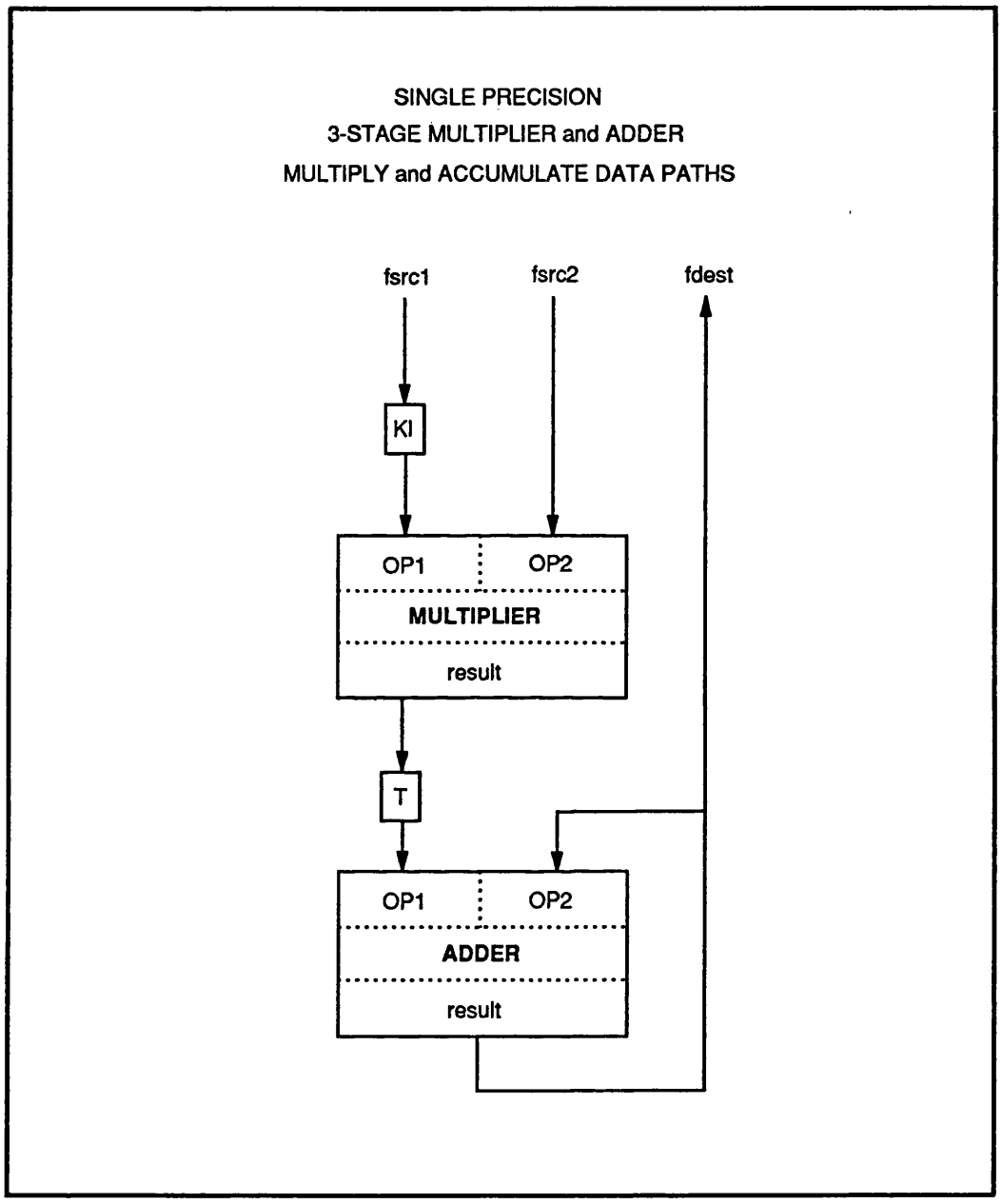

Figure 7.5: Data paths for multiply and accumulate.

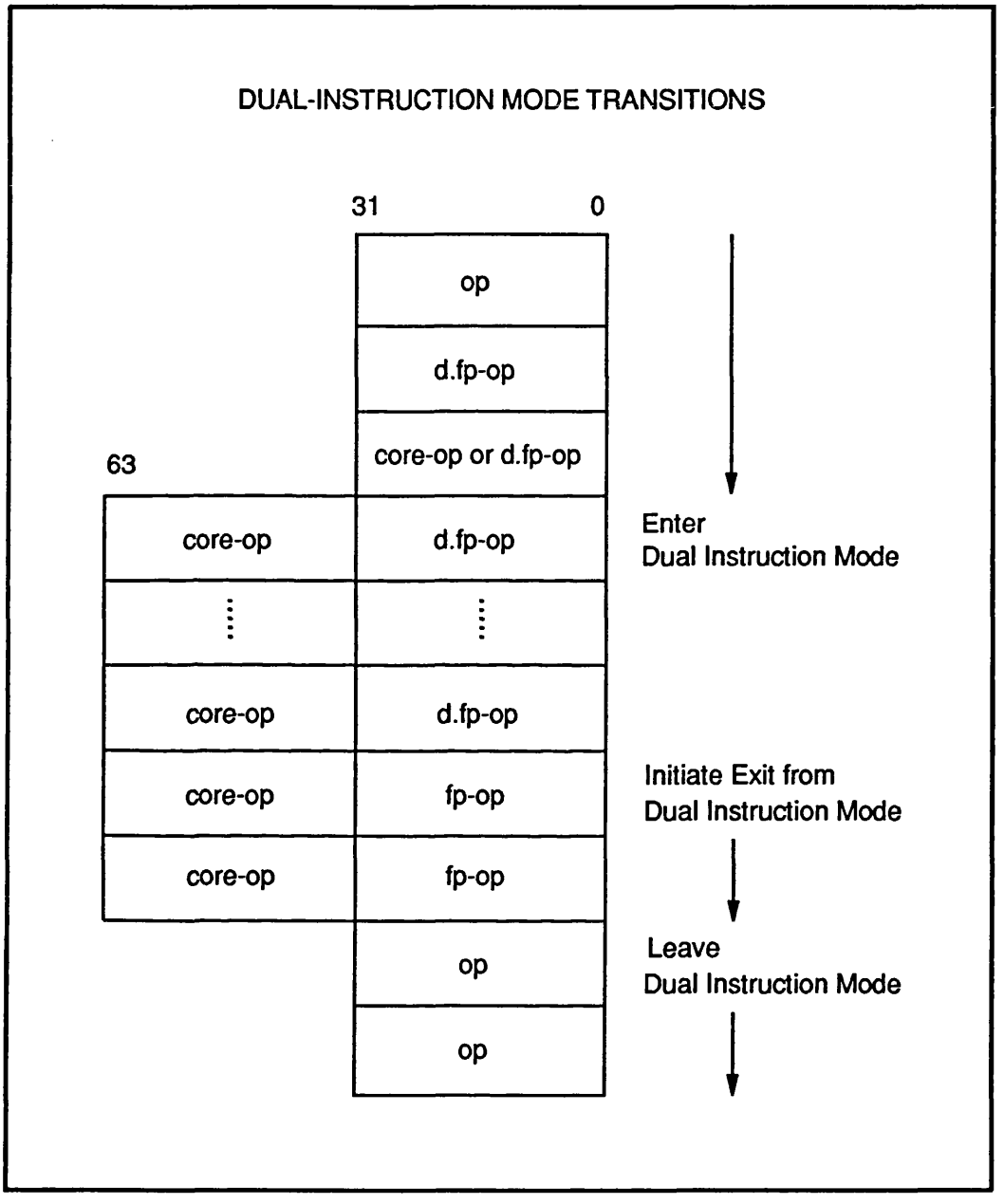

Figure 7.6: Dual-instruction mode transitions

Chapter 8

System data and simulation results

8.1 Introduction

This chapter validates and discusses the simulation results obtained from the newly developed simulators. In all the simulation studies, a frequency dependent distributed transmission line model as described in Chapter 4 is used. A simplified source model, using a constant voltage behind subtransient impedance, is used throughout the modelling of all sequences involving the electromagnetic transient studies. In addition to this, an actual power system source model is also used in the EMTP simulation studies in order to study the low frequency electromechanical transients.

Results obtained from the new simulators are verified through a comparison with those generated from the traditional mainframe simulators. Discussions on the computational time for achieving the 'real time' target are given for the new simulators.

8.2 System circuits studied

Line data: Each line is a typical quad conductor 400kV line, 128km in length (unless otherwise specified) and has the following line data:

- the quad conductor consists of $4 \times 54/7/0.33cm$ s.c.a. with $0.305m$ bundle spacing,
- earth wire is $54/7/0.33cm$ s.c.a., and
- earth resistivity is $100\Omega m$.

Single conductor system: All the simulation studies for a single conductor system are based on the line configuration as shown in fig. 8.1. The transmission system studied is a double end fed single conductor system as shown in fig. 4.1.

Three phase system: All the simulation studies for a three phase system are based on the line configuration as shown in fig. 8.2. The transmission system studied is a double end fed untransposed three phase system as shown in fig. 4.2.

Fault sequences: The results presented here relate to the following fault sequences (unless otherwise specified):

- a fault inception takes place somewhere on the line at time t_{fault},
- the circuit breaker poles at both ends of the line are opened *simultaneously* at time t_{open},
- the fault is cleared at time t_{clear},

- the circuit breaker poles at both ends of the line are reclosed *simultaneously* at time $t_{reclose}$.

8.3 Time convolutional modelling

General considerations: It should be noted that a series of simulation tests have shown that the total number of coefficients used in the numerical calculation of the convolutional integral (eqn. 6.7) must be an integer multiple of the number of coefficients per power frequency period. For example, given that the sampling frequency is 4000Hz and the power frequency is 50Hz, the number of coefficients per period should be 80. Thus, the number of coefficients used in the evaluation of the convolutional integral must be $80n$, where $n = 1, 2, 3, \cdots$, etc.

Simulation results: A series of simulation results were generated from the newly developed simulator by using a different number of coefficients for solving the convolutional integrals. Comparisons were then made with the results attained from the well tested mainframe computer simulations developed by Johns and Aggarwal [1, 2]. With reference to the single conductor system as shown in fig. 4.1, with line length of $128km$ and sending and receiving end capacities of 5GVA and 35GVA respectively, the simulation study relates to an earth fault at $85km$ from the sending end at time $t_{fault} = 0.025s$, i.e., at near maximum voltage. The fault path impedance is 1Ω, and the sampling frequency is 4000Hz.

- Fault transient voltages:

  Fig. 8.3a represents the sending end voltage obtained using 80 coeffi-

cients. It can be seen that the voltage waveform (fig. 8.3a) matches closely with the corresponding frequency domain results shown in fig. 8.3b. However, it should be observed that there are very small differences present, particularly with the magnitudes of the initial high frequency transients. An increase in the number of coefficients has little effect on the accuracy attained as evident from figs. 8.3c to 8.3f.

- Fault transient currents:

  With regard to the fault transient current waveforms obtained with convolutional technique, a comparison of figs. 8.4a and 8.4b clearly shows that the sending end current obtained using the convolutional technique corresponds closely to that obtained using Johns and Aggarwal's frequency domain model for only about two cycles following a fault, with a significant reduction in magnitude subsequently in the case of the former. An increase in the number of coefficients to 240 and 400 extends the number of periods for which the current waveforms obtained using the convolutional technique closely corresponds to that obtained using the frequency domain model, as evident from figs. 8.4c and 8.4d. This phenomenon in the case of the current waveforms can best be explained by considering the behaviour of the impulse response of the admittance (Y_D) which is used for evaluating the sending end current (eqn. 6.13).

- Impulse response functions for voltage and current calculation:

  Fig. 8.5 shows the impulse response of the expression ($Z_{ss}Y_D$) which is used for the sending end voltage calculation (eqn. 6.15). Fig. 8.6 shows the impulse response of the admittance (Y_D) which is used for the sending current calculation (eqn. 6.13). The characteristics of the

impulse response functions show that in order to accurately evaluate the convolutional integral, the total number of coefficients used must be chosen up to the point in time where there is very little energy left in the impulse response function. With reference to the impulse response function used for voltage calculation shown in fig. 8.5, two important points can be observed. Firstly, the impulse response function oscillates (more or less symmetrically) along the zero value axis. Secondly, it rapidly converges towards zero within one to one and a half power frequency cycles. This effectively means that even with 80 coefficients, which is equivalent to one power frequency cycle, the sending end voltage can be calculated with reasonable accuracy. It should be recalled that the numerical approximation for the convolutional integral is given by eqn. 6.7, i.e., an output point is yielded by summing all the multiplication results, which are obtained by multiplying together the data points and their corresponding coefficients.

Inspection of the impulse response of the admittance variable (Y_D) (fig. 8.6) used in the calculation of the sending end current reveals that the impulse response function converges towards zero very slowly and starts to diverge towards the end. More importantly, the impulse response function does not oscillate along the zero value axis, but rather along a negative value trajectory. These characteristics of the impulse response function explain the reasons for the inaccuracy in the sending end current obtained as described previously. As can be seen from fig. 8.6, even with 400 coefficients (which is equivalent to five power frequency cycles), there is still a large amount of energy left in the impulse response function. This thus leads to a very large truncation error in the approximation of the convolutional integral cal-

culation.

Overall discussion: The results obtained show that the time convolutional method is prone to producing large unacceptable errors in approximating the convolutional integrals using a small number of coefficients, particularly in the case of currents. A very large number of coefficients are required for the currents to achieve a reasonable degree of accuracy. For example, even with 400 coefficients, as can be seen in fig. 8.4d, the sending end current, I_s is quite different from that obtained from the frequency domain technique as shown in fig. 8.4b. It has also been found that, in general, the impulse response of each of the element in the system Y-matrix (eqn. 4.21) has the characteristics of converging towards zero very slowly (e.g., figs. 8.7 and 8.8 show the impulse responses of the Y-matrix elements, Y_B and Y_C respectively).

The results have shown that in terms of accuracy and hence of computational time, the time convolutional technique has no advantage over the frequency domain technique. In view of this finding, it was thus decided not to investigate this technique any further.

8.4 Frequency domain modelling

General considerations: It should be noted that in order to perform realistic simulation and hence obtain realistic computational time measurement, the following constraints must be considered:

1. the total number of samples to be used in the simulation,

2. the sampling frequency, and

3. the transient observation time.

In order to use the FFT and IFFT routines effectively, the number of simulated samples used must equal to 2 to the power of n, i.e., the total number of samples, N, used for the transient calculation must be:

$$N = 2^n$$

where $n = 1, 2, 3, \cdots$, etc. However, the larger the number of samples used, the longer it will take to calculate the transients, i.e., the larger the computational time will be. Hence, a sensible choice of simulated samples must be decided. From a practical point of view, for circuit breaker pole opening transients, a minimum of 512 samples with a minimum sampling frequency of 4000Hz, giving a transient observation time of $128ms$, is required. It was therefore decided that for benchmark computational time measurement, all the transient calculations should be based on 512 samples and 4000Hz sampling frequency.

Multiprocessing strategies: Chapter 4.2 derives the following system equations for the fault transient simulation of a basic double end fed transmission system (fig. 4.1 and 4.2):

$$\begin{bmatrix} I_f \\ I_s \\ I_r \end{bmatrix} = \begin{bmatrix} Y_A & Y_B & Y_C \\ Y_D & Y_E & Y_F \\ Y_G & Y_H & Y_I \end{bmatrix} \begin{bmatrix} E_{ff} \\ E_s \\ E_r \end{bmatrix} \tag{8.1}$$

$$\begin{bmatrix} E_{ff} \\ E_s \\ E_r \end{bmatrix} = \begin{bmatrix} Z_A & Z_B & Z_C \\ Z_D & Z_E & Z_F \\ Z_G & Z_H & Z_I \end{bmatrix} \begin{bmatrix} I_f \\ I_s \\ I_r \end{bmatrix} \tag{8.2}$$

$$V_s = E_s - Z_s I_s \tag{8.3}$$

$$V_r = E_r - Z_r I_r \tag{8.4}$$

It has been found that about 50 per cent of the computational time for the transient calculation is spent on performing Fourier and Inverse Fourier transformations for transforming the data into and out of the frequency domain. An extensive series of investigations into different multiprocessing strategies for the fault transient simulation have shown that the best suitable technique to achieve optimum performance is to use one processor to calculate one system variable of interest. This has the great advantage of multiprocessing the FFT and IFFT routines for each variable of interest while balancing the cost of communication overheads among the processors. As defined in eqns. 8.1 and 8.2, for a single conductor system, there are altogether six such system variables, namely, I_f, I_s, I_r, E_{ff}, E_s and E_r. For each event of the fault sequence, these six system variables must be solved. Thus, using one variable one processor strategy, six processors are required, i.e., one for I_f, one for I_s, and so on. For example, with reference to fig. 7.3, which shows a node of four T800 transputers, processor 'slave-1' can be assigned to calculate the fault transients of the system variable, I_s, while processor 'slave-2' can be used for I_r, and processor 'slave-3' is for V_s. It should be noted that the sending end voltage V_s is simply related to the circuit breaker pole voltage E_s by equation 8.3. There are thus five time-domain opening transients, namely, V_s, I_s, V_r, I_r and I_f which must be produced in 'real time'. The first four are input signals to the relay and the last one is required for the subsequent fault break-off simulation. Since E_s is not needed to be transformed straight back into the time domain for the breaker pole opening stage, V_s and E_s can be calculated in the same processor. Similarly, V_r and E_r are simply related by equation 8.4 and can also be calculated in the same processor. However, the time domain opening transients of E_s and E_r are required for the subsequent evaluation of the fault clearing and reclosure transients. The process of transforming

the opening transients of E_s and E_r from the frequency domain back to the time domain is thus postponed to the fault clearing stage. While waiting for the injected fault path current I_f to be formed in the processor for calculating I_f from the time domain to the frequency domain, the opening transients of E_s and E_r can be transformed back to the time domain. A similar technique can likewise be applied to the calculation of other variables, i.e., interleaving the transient calculation and the stages of the fault sequence and keeping the processor in constant and productive use.

For a three phase system (fig. 4.2), each element of the system Y-matrix and Z-matrix of eqns. 8.1 and 8.2 is a 3 by 3 submatrix, and variables such as I_f, I_s, I_r etc. are 3 by 1 vectors, one for each phase. Thus, there are altogether 18 system variables of interest for a three phase system and 18 processors are required.

Computational time measurement strategy: Chapter 6 elaborates on the transient calculation techniques. In this section, the analysis of the circuit breaker pole opening transient calculation is extended by considering the amount of calculation involved in solving each variable of interest. This thus ascertains the computational time measurement strategy to be used.

For the purposes of illustrating the overall concepts, a single conductor system is considered and the concept can be extended to a more complicated three phase situation. For the breaker pole opening stage, three system variables, namely, E_{fft2}, I_{st2} and I_{rt2} are known, i.e., $E_{fft2} = 0$, and I_{st2} and I_{rt2} are the injected currents. The remaining three system variables, namely, I_{ft2}, E_{st2} and E_{rt2} are to be solved from the system equations 8.1

and 8.2. It should be noted that V_{st2} is related to E_{st2} by equation 8.3 and can only be obtained after E_{st2} has been found. Similarly, V_{rt2} is related to E_{rt2} by equation 8.4 and can only be obtained after E_{rt2} has been found. It can thus be seen that, as far as the circuit breaker pole opening stage is concerned, the amount of computation involved in obtaining the variable, V_{st2} is the largest. This means that the transient calculation time for the latter is the longest, and the transient calculation time for V_{rt2} will be the same as that for V_{st2}. It can thus be seen that for investigation purposes, in order to obtain a calculation time which can be served as a representative for the worst case scenario, for the single conductor system, the benchmark calculation time should be based on the time measurement of evaluating the opening transients of either the sending end or the receiving end voltage. In this work, the sending end voltage, V_{st2}, has been chosen as a representative for benchmark calculation time measurement.

For a three phase system, it can be shown that the amount of computation for each of the sending voltages (V_{sa}, V_{sb} and V_{sc}) and the receiving end voltages (V_{ra}, V_{rb} and V_{rc}) are the same. And, amongst other system variables, the amount of computation involved in obtaining each of the above six system variables is the largest. It can thus be seen that the calculation time measurement should be based on the time measurement of evaluating the opening transients of any one of the sending end or the receiving voltages. In this work, the sending end phase-a voltage, V_{sat2}, has been chosen as a representative for benchmark calculation time measurement.

Simulation studies:

1. **Single conductor line study**

Validation of results: A series of simulation studies, such as using different combinations of sending end and receiving end source capacities, different fault locations and different fault instants (detailed effects of these variations are described later in the more general three phase simulation studies), have shown that in every study case, the waveforms obtained from the newly developed simulator are effectively identical to those generated from the mainframe simulators by Johns and Aggarwal [1, 2]. Fig. 8.9 represents the sending end voltage and current for a voltage maximum fault. Referring to fig. 8.9a, it can be seen that after fault inception, the sending end voltage is significantly distorted and the voltage magnitudes are significantly reduced. However, several large peak values are observed during the initial period after the fault inception. After the opening of the breaker poles at both ends, significant transient distortions are again observed, and these transients gradually die down to zero level. Furthermore, during the fault break off and up to the time of reclosure, the sending end voltage is very small in magnitude. During the initial period of breaker reclosure, overvoltages and transient distortions are observed for about one to two cycles. However, the transients eventually die down and the voltage returns to normal steady state values.

Referring to fig. 8.9b, it can be seen that after fault inception, high fault sending end current is built up. However, the current waveform is nearly distortion free. After opening the breaker poles, the current collapses to zero. On reclosure, before returning to the normal steady

state, significant transients are observed.

Referring to fig. 8.10a, it can be seen that the fault path voltage collapses to near zero on fault inception. The voltage across the transient fault path subsequently recovers to a normal value after the reclosure of the breaker poles at both ends. The corresponding fault path current is shown in fig. 8.10b. It can be seen that a high fault path current is developed after fault inception. After the opening of the breaker poles, the current collapses gradually to a very small value. Very small ripples are due to the trapped charge in the circuit. However, these ripples disappear after fault break-off.

Computational time measurement: Table 8.1 shows the computational time measured by running the simulator on T800 transputers. It should be noted that the term *'optimizations'* in this context corresponds to the hand-made modifications to the simulating source codes. Detailed descriptions of the hand-made 'optimizations', which have been done to the simulating codes, can be found in Chapter 7.5. Three

Before optimizations	After optimizations	Multiprocessing
8.15s	1.17s	0.60s

Table 8.1: V_s computational time on T800 transputers: simulation time = $0.128secs$, samples = 512, sampling freq. = $4kHz$

set of results are shown for measuring the calculating time to obtain the opening transients of the sending end voltage, V_s. The first set shows the calculating time obtained using the modified simulator without optimizations. The second set is obtained by running the modified and optimized simulator. It should be noted that the first two sets were obtained by running the simulators in a single T800 transputer.

The third set is obtained by running the multiprocessing simulator on a node of 4 transputers, using the 'one processor one variable strategy'. It should be noted that in order to obtain the opening transients of the sending end voltage, V_s, three processors are required, i.e., one processor for I_s to calculate the injected sending end current, one processor for I_r to calculate the injected receiving end current, and one for V_s to manipulate the opening transients of the sending end voltage. Referring to table 8.1, it can be seen that an improvement in speed by a factor of 14 has been achieved through the optimizations and multiprocessing. The reasons that it is not quite three times faster through multiprocessing (when compared to the second set) are, firstly, because there are communication overheads among the processors and secondly, the amount of calculation for obtaining opening transients of V_s, as explained before, is larger than that for obtaining either I_s or I_r. It should also be noted that this calculating time will remain more or less the same even if all the other system variables are to be computed concurrently. This is so because by adopting the 'one processor one variable strategy', the amount of calculation in the processor for calculating V_s remains the same. It should also be mentioned that the effects of fault location, fault instant, or any changes in the sending end and receiving end capacities will not affect the computational time, since the amount of computation for each variable of interest remains essentially the same.

2. Three phase line study

The results obtained from the single conductor simulation were very encouraging. It was therefore decided to extend the study to a three phase system.

Validation of results: Throughout all the simulation tests, it has been found that the results obtained from the newly developed simulator are effectively identical to those obtained from the mainframe simulators by Johns and Aggarwal [1, 2]. To illustrate the validation of the results, the following study cases are performed:

- **Effect of source parameters:**

  The source parameters, particularly their capacities, significantly affect the fault transient waveforms. Figs. 8.11 and 8.12 show the comparison between the system responses by taking the sending end source capacity with large and small values. It can be seen that the voltages (fig. 8.11) are much smoother when the local source near the point of observation is large. Such a response is obtained because the travelling wave components of current propagated into the source do not cause significant voltage transients if the source impedance is small, i.e., the busbar voltage is held nearly constant and is not easily distorted. It is of interest to note that in the case of a small sending end source capacity, very considerable waveform distortion occurs. This is due to the low capacity of the source adjacent to the point of observation, which therefore constitutes a major point of electrical discontinuity from which high-frequency components are easily reflected. Very considerable distortion is observed in the sound phases.

- **Effect of fault types:**

  Faults not involving earth give rise to waveforms which are generally very distorted. Fig. 8.13a shows the voltages obtained for an a-b phase fault, and by comparing this with an 'a'-earth for corre-

sponding source conditions, shown in fig. 8.11a, it is clear that the travelling waves persist for considerably longer in the former case. Persistence of the travelling waves is due to the lower attenuation of the aerial modes associated with phase faults. The sound phase voltage does not contain significant travelling wave components, due to the nearly equal and opposite nature of the current which flows in the faulty phases (fig. 8.13b).

- **Effect of fault instant:**

  In each of the foregoing studies, the faults have been applied at an instant corresponding to voltage maximum in the faulty phase or phases. These are the worst cases from the point of view of travelling wave distortion. The other extreme is reached when the fault is applied at zero voltage. In this case, the travelling waves are reduced because there is not a large and sudden voltage change at the point of fault. Fig. 8.14 shows the waveforms for a fault at voltage zero at the receiving end. Distortion is extremely small, and the well known offset nature of the current waveform is clearly observed.

- **Effect of fault position:**

  When the fault position becomes more distant from the observation point, the frequency of the superimposed components decreases due to essentially the transit time between the source discontinuities and the fault point increasing. This phenomenon can be verified by comparing the sending end voltage waveforms shown in fig. 8.11a for a fault which is closer to the sending end, and fig. 8.15a for a fault which is near the receiving end. It can be

seen that although the magnitudes of the currents increase as the fault distance decreases (due to a reduction in impedance), the transients on the current waveforms are more or less the same. This can be verified by comparing the sending end current waveforms shown in fig. 8.12a for a fault which is closer to the sending end, and fig. 8.15b for a fault which is near the receiving end.

- **Simulation of full autoreclosure sequence:**

  - Three phase autoreclosure:

    Fig. 8.16 shows the sending end voltage responses for a three phase autoreclosure sequence for a mid-point three phase to earth fault at voltage maximum of phase-a. Fig. 8.17 shows the corresponding current responses, and figs. 8.18 and 8.19 represent the fault path voltages and currents respectively.

    Referring to fig. 8.16, it can be seen that after fault inception, the sending end voltages decrease in magnitudes, which is as expected since the fault paths effectively act as potential dividers. Maximum transient distortions are observed on phase-a. This is because the fault instant for phase-a is at voltage maximum. Phase-b and c voltages are also quite significantly distorted, but the distortion is much less when compared to that of phase-a. After the opening of the breaker poles, very large transient distortions with very large magnitudes (at least during the initial period subsequent to breaker pole opening) are observed on all the three phases. This is because the circuit breaker poles are opened simultaneously,

rather than sequential pole opening taking place near current zero for each phase, and this leads to current chopping on opening the breaker poles. Since the fault is a three phase to earth fault, there is no significant trapped charge in the line, and the transients thus gradually die down to near zero level. Furthermore, during the fault break off and up to reclosing time, the sending end voltages are very small in magnitude. At this point, it must be mentioned that if the fault type is other than three phase to earth fault, there will be significant amounts of trapped charge in the unfaulted phase(s) after the opening of the breaker poles, and a dc voltage level will be sustained in the unfaulted phase(s). This phenomenon can be observed in figs. 8.20 to 8.23 which show the responses of other fault types. It should be noted that the magnitude of the dc voltage level will depend very much on the pole opening angle of the phase concerned, e.g., the level will be near zero for near voltage zero pole opening, and will be a maximum for voltage maximum pole opening. During the initial period of breaker reclosure, overvoltages and transient distortions are observed on all the phases for about one to two cycles. However, the transients eventually die down and the voltages return to normal steady state values.

Referring to fig. 8.17, it can be seen that after fault inception, large sending end currents are sustained on all the phases. However, the current waveforms are nearly distortion free. After breaker poles opening, the three currents collapse to zero. On reclosure, significant transients are observed on the three

currents and these gradually die down and the system returns to normal steady state.

Referring to fig. 8.18, it can be seen that the fault path voltage collapses to near zero on fault inception. Furthermore, it can be seen that the voltage levels in all the three phases stay near zero. This is so by virtue of the fact that since this is a solid three phase to earth fault, the residual current on breaker opening does not cause any significant voltages to develop across the fault path. The voltages across the transient fault path, subsequently recover to normal values after the reclosure of the breaker poles at both ends. The corresponding fault path currents are shown in fig. 8.19. It can be seen that large fault path currents are developed after fault inception. After the opening of the breaker poles, the three currents collapse gradually to very small values. Very small ripples are due to the trapped charges in the circuit. However, these ripples disappear after fault break-off.

- Single pole autoreclosure:

  Fig. 8.24 shows the sending end voltage responses for a single pole autoreclosure sequence for a mid-point phase-a to earth fault. Fig. 8.25 shows the corresponding sending end current responses. It can be seen that similar transients, as observed on the faulted phase waveforms for the three phase autoreclosure, can also be found in phase-a. The sound phase voltages are nearly distortion free throughout the sequence. However, after fault inception, slight overvoltages are observed on the

sound phases, and the sound phase currents are much larger than their corresponding normal steady state values.

Computational time measurement: It should be mentioned that the new simulators were developed and optimized at different stages. For example, the prefault steady state and the fault inception stages were firstly developed. Modifications and hand-made optimizations were then carried out to these codes before the development of the breaker pole opening stage was launched. Furthermore, some routines, which had already been modified and optimised in the development of the single conductor system simulation, were directly adopted in the simulation of the three phase system. As a result, in the process of developing the simulation codes for the breaker pole opening stage, and later on the subsequent fault clearing and breaker pole reclosure stages, some routines, e.g., the FFT and IFFT routines, which had been optimised in the previous stages, were directly adopted. Since the source codes of the traditional mainframe simulator cannot be accessed, it is quite difficult to compare the overall calculation time for each event of the fault sequence (e.g., for breaker pole opening stage, in particular) between the unoptimized mainframe simulator and the new optimized simulator.

However, it must be emphasised that amongst other hand-made optimizations (see Chapter 7.5), the following two optimizations have been proven to have significant influence on the calculation time for the circuit breaker pole opening stage. The first one is the modified and optimized FFT and IFFT routines. The second one is the pre-calculation of the frequency spectra of submatrices and inversion

of submatrices. Investigation into the first one has shown that the modified and optimized FFT and IFFT routines (see Chapter 7.5 and Appendix C) are about four times faster. For the second one, table 8.2 shows the computational time comparison between the simulators with and without the pre-calculation of the frequency spectra of the submatrices by running the simulators on a single T800 transputer. Two set of results are shown for measuring the calculating time to obtain the opening transients of the sending end phase-a voltage, V_{sa}. The first set shows the calculating time obtained using the modified simulator without the optimizations. The second set is obtained by running the optimized simulator. It can clearly be seen that there is a significant

Fault type	Before optimizations		After optimizations	
	Single Pole	Three Phase	Single Pole	Three Phase
Phase-a to earth	7.91s	18.97s	0.48s	0.98s
Phase-ab to earth		16.36s		0.98s
Three phase to earth		14.0s		0.96s

Table 8.2: V_{sa} computational time on a single T800 transputer: simulation time $= 0.128secs$, samples $= 512$, sampling freq. $= 4kHz$

improvement in the speed after the optimizations.

Table 8.3 tabulates the computational time for the sending end voltage, V_{sa}, for opening the circuit breaker poles by running the newly developed simulator on an $i860$ microprocessor. It is assumed that for each processor, one variable, in this case, V_{sa}, is to be calculated. Table 8.4 shows the computational time obtained by running the program on transputers. Each computational time includes:

(a) Six forward FFT routines in case of three phase autoreclosure, or two forward FFT routines in case of single pole autoreclosure.

(b) Frequency domain transient manipulations to get V_{sa}.

(c) One backward FFT routine which transforms V_{sa} from the frequency domain back to the time domain.

Tables 8.3 and 8.4 clearly show that the type of hardware used affects the computational time very dramatically. It can be seen that the optimized program running on a single Intel i860 processor is about twenty four times faster than running on a single Inmos T800 transputer.

For three phase autoreclosure, it has been found that the computa-

Fault Type	Single Pole Auto.	Three Phase Auto.
Phase-a to earth	20ms	40ms
Phase-ab		40ms
Phase-ab to earth		35ms
Three-phase to earth		40ms

Table 8.3: V_{sa} computational time running on a single i860 processor: simulation time = $0.128secs$, samples = 512, sampling freq. = $4kHz$

Fault Type	Single Pole Auto.	Three Phase Auto.
Phase-a to earth	480ms	980ms
Phase-ab		980ms
Phase-ab to earth		960ms
Three-phase to earth		980ms

Table 8.4: V_{sa} computational time running on a single T800 transputer: simulation time = $0.128secs$, samples = 512, sampling freq. = $4kHz$

tional time for six forward FFT routines is $30ms$. This implies that $5ms$ is required to perform each FFT routine. The computational time for one backward FFT routine is $2.5ms$. Thus, $7.5ms$ is required to calculate one forward and one backward FFT routine. Hence, if one processor for each variable were to be used, then $15ms$ [1] would be required to calculate each variable of interest. The overall calculation time for

[1]Since the six forward FFT routines can be calculated in six processors concurrently, therefore, the computational time for V_{sa} would be $40ms - 30ms + 5ms = 15ms$.

each variable would be $15ms$ plus the communication overhead.

It can thus be seen that for each variable of interest, the computational time for the worst case is $15ms$ for a simulation time of $128ms$ with 512 simulation samples and a time step length of $250\mu s$. It should be remembered that the frequency domain technique requires to calculate the whole set of simulation samples, in this case 512 samples, before the first useful new output sample, which can be used as an input to the relay, is obtained. Thus the first new output sample can only be obtained after $15ms$ of transient computation for each circuit change.

For single pole autoreclosure, the computational time for two forward FFT routines is $10ms$. Hence, if one processor for each variable were to be used, then $15ms$ [2] would be required to calculate each variable of interest. Again, the overall calculation time for each variable would be $15ms$ plus the communication overhead.

It should be recalled that with reference to modern high speed protection relays, coupled with fast switchgear, from the time that a tripping signal is sent by a relay to open a circuit breaker, to the time that the circuit breaker contacts start to separate, is typically around $40ms$. Within this $40ms$, the simulator must be able to do two things. Firstly, from the fault inception transient waveforms, which have been calculated off-line, it must be able to locate the exact times of the current zero crossings of the respective phases. It is at these times that the circuit breaker poles of the respective phases are to be opened. Secondly, it must be able to calculate all the circuit breaker pole opening

[2]Since the two forward FFT routines can be calculated in two processors concurrently, therefore, the computational time for V_{sa} would be $20ms - 10ms + 5ms = 15ms$.

transients.

From the computational time measured, it can be seen that it is possible to simulate the transients and test a relay interactively in real time for single pole autoreclosure even if the circuit breaker poles are opened at different times. However, from a practical point of view, for three phase autoreclosure, it is only possible to simulate the transients and test a relay interactively in real time provided that the circuit breaker poles are opened simultaneously in the simulation test. It may require a large number of Intel i860 microprocessors, which is economically not viable, to achieve the real time requirement for sequential pole opening.

Hardware requirements: Currently, the simulator is being developed under a single Intel i860 processor. From the above computational time, it can be seen that in order to achieve the 'real time' target, one i860 processor for calculating one or several variables is suggested.

In this respect, it should be mentioned that, in order to initiate circuit breaker opening and any other subsequent circuit changes, there is a requirement to calculate a maximum of 18 variables at any one instant. This would effectively mean that in the worst case scenario, 18 processors would be required. However, it may be possible that with a reduced number of samples, two variables could be calculated on one processor in the required time, thus reducing the number to 9 processors.

8.5 EMTP modelling

1. **Simplified source model:**

   Validation of simulation results: A series of tests have shown that the results obtained from the newly developed simulator have close correspondence to those generated from the frequency domain simulator. In order to illustrate and validate the results, the following simulation studies for different fault types are performed. Results obtained are plotted on the same graph with those obtained from the frequency domain simulator (FDS) [1, 2]. It should be noted that the faults are applied at the midpoint of the transmission line and the prefault phase-a voltage is near maximum. Figs. 8.26 and 8.27 show the sending end voltages and currents comparison for a phase-a to earth fault. Figs. 8.28 and 8.29 show the comparison for a double phase (phase-a and b) to earth fault. Figs. 8.30 and 8.31 show the comparison for a pure inter-phase (phase-a and b) fault. Figs. 8.32 and 8.33 show the comparison for a three phase to earth fault. Figs. 8.34 and 8.35 show the comparison for a phase-a to earth near voltage zero fault. With reference to the above figures (figs. 8.26 to 8.35), it can be seen that the results obtained from the new simulator have close correspondence to those obtained from the frequency domain simulator.

   In order to illustrate the breaker pole opening transients and full fault autoreclosure sequence, the following simulation studies are performed. Figs. 8.36 and 8.37 illustrate breaker pole opening transients for a double phase (phase-a and b) to earth fault with the circuit breaker poles at both ends being opened simultaneously. Figs. 8.38 and 8.39 show the opening transients for the same fault conditions as the previous

two figures (8.36 and 8.37) except that the breaker poles at both ends are sequentially opened at their current zeros. It can be seen that the opening transients of the sending end voltages are significantly reduced for the latter case. Furthermore, it can be observed that since the poles are opened at current zeros for the latter case (i.e., at different instants when compared to the former case), the sound phase sending end voltage (phase-c) sustains a different dc-voltage level after the opening of the breaker poles. Figs. 8.40 and 8.41 show a three phase autoreclosure sequence for a midpoint phase-a to earth fault with sequential breaker pole opening at both ends.

It should be noted that in general, similar observations, as discussed previously in section 8.4 for frequency domain modelling, can also be found on the waveforms of the above EMTP simulation studies. The reasons for the discrepancies between the waveforms produced by the two simulators are mainly due to, e.g., the constant transformation matrix approximations in the time domain method and the differences in the numerical solutions between the two techniques which have been described in the previous chapters.

Discussion: It should be noted that, due to the modelling technique used, the time step length used must equal to or be smaller than the trav-

Line length (km)	5	25	64	75	100	160
Time step length (μs)	15	80	210	250	330	530

Table 8.5: Largest time step length which can be used for the corresponding fault distance.

elling time of the fastest wave from the sending end to the fault point or from the receiving end to the fault point, whichever is smaller of

the two. This means that the location of fault will affect the choice of time step length. Table 8.5 shows the largest time step length which can be used for various fault locations measured from the sending end or the receiving end. It can be seen that if the faults are close to either the sending end or the remote receiving end, a very small time step length must be used. This is detrimental from the 'real time' simulation point of view, because with the same amount of computation for each time step, the smaller the time step length, the larger is the burden imposed on the computational power and speed.

However, it must be emphasised that if the faults are very close to either the sending end or the receiving end, the time step length chosen can be equal to or smaller than the travelling time of the fastest wave from the sending end to the receiving end. Thus, these two extremes can be used to test the relays for faults close to either the sending end or the receiving end and the 'real time' burden can be significantly reduced.

2. **Actual generator model:**

- **Simulation data:**

  Referring to a single machine power system model as shown in fig. 5.1, the following system data are used:

  - Steady state power frequency: 60Hz.
  - Synchronous generator:
    (a) Ratings: 892.4MVA, 2 poles, 26kV.
    (b) Air-gap line (AGL) and saturation curve:

1800A Field current which produces rated voltage on the AGL.

1907A Field current which produces rated voltage on the no-load saturation curve.

3050A Field current which produces 1.2pu voltage on the no-load saturation curve.

(c) Electrical data: The Canay's characteristic reactance [5, 6] is not known. It should have the same value as that of the leakage reactance, X_l.

$R_a = 0.0pu$;

$X_d = 1.790pu$; $X_l = 0.130pu$

$X'_d = 0.169pu$; $X_q = 1.710pu$

$X''_d = 0.135pu$; $X'_q = 0.228pu$

$\tau'_{do} = 4.30s$; $X''_q = 0.20pu$

$\tau''_{do} = 0.032s$; $\tau'_{qo} = 0.850s$

$X_0 = 0.130pu$; $\tau''_{qo} = 0.050s$

(d) Mechanical data: The inertia constant, $H = 4.5$ sec.

- Excitation system: The constants used are shown in table 5.1 with values corresponding to the rotating rectifier exciter.
- Governor: The constants used are shown in table 5.2.
- Generator transformer: The transformer is Delta-Star (26/400kV) connected. It has been assumed that there are no losses and the reactance value is in per unit on the machine base power. The total reactance, $X_{tx} = 0.100pu$.
- Transmission line: Line length is 320km. The fault location is 160km from the sending end.

- Ideal source: The ideal source capacity is about 5GVA.
- Fault path impedance: The fault path impedance is 1Ω.

- **Validation of simulation results:**

  A series of simulation tests have shown that similar phenomena, as found from the frequency domain simulator [41], can also be observed from the new simulator. The following study cases have been chosen to enable the salient features of the system faulted responses to be illustrated and to enable the validity of the new simulator developed.

  - **Phase-a to earth fault:**

    Fig. 8.42 shows the voltages at the machine terminals and the sending end for a midpoint phase-a to earth fault which occurs at approximately 54 degrees before the positive to negative zero-crossing of the fault point voltage (fault inception time = $22.5ms$). The corresponding current waveforms are given in fig. 8.43. With reference to fig. 8.42, the delays associated with the propagation of the travelling waves are clearly evident. Furthermore, it can be seen that earth mode components of travelling wave voltages do not propagate into the machine and that the phase-b voltage does not contain high frequency distortion. This is as expected because earth mode current components, being synonymous with zero sequence components, are effectively trapped and short circuited by the delta winding of the generator/transformer and do not establish corresponding components of voltage at the machine terminals. With reference to fig. 8.43, it can be seen that phase-a current is strongly offsetted due to the transient exponential dc components.

- **Double phase to earth fault:**

  Figs. 8.44 and 8.45 respectively show the voltages and currents for a double phase to earth fault which occurs near the voltage maximum of the line side phase-a voltage. It can be seen that the generator and its transformer clearly constitute a major point of discontinuity from which travelling waves are readily reflected. For this particular fault condition, the aerial modes of the travelling wave voltage are of a large and almost equal and opposite magnitude, and since they are additive in the c-phase, the sound phase voltage on the line side of the generator is relatively distortion free. On the other hand, the large valued aerial mode components give rise to significant overvoltages on the faulted phases.

- **Pure interphase fault:**

  Figs. 8.46 and 8.47 show the voltages and currents for a pure interphase (b-c) fault which occurs near the zero of the prefault voltage between the faulty phases. With reference to fig. 8.47, it can be seen that complete offsetting of the a-phase current at the sending end of the line and the machine terminals is produced. The healthy a-phase voltage on the line side of the transformer, shown in fig. 8.46, is more or less travelling wave distortion free due to the fact that equal and opposite travelling wave components of fault current in the faulted phases are produced for this particular type of fault.

- **Comparison with the ideal source model:**

The following simulation studies demonstrate some selected simulation

studies, which illustrate the comparison between the system responses obtained using the actual generator source model (fig. 5.1) and the results obtained using the simplified source model (fig. 4.2). It has been assumed that the breaker poles are opened at current zeros of the respective phases.

- **Phase-a to earth, near voltage maximum fault:**

  Figs. 8.48 and 8.49 show the sending end voltages and currents comparison between the two models for a midpoint phase-a to earth near voltage maximum fault. With reference to the above two figures, it can be seen that the system responses obtained using the actual generator model are very similar to those obtained using ideal source model. The faulted phase current (fig. 8.49) obtained using the two models is very much the same. However, it should be observed that the sound phase currents obtained using the actual generator model have small fluctuations. Figs. 8.50 and 8.51 show the extended simulation of the sound phase currents (b and c) with a longer observation time. With reference to the above two figures, small fluctuations of sound phase currents (phase-b and c) are clearly observed after the pole opening of the faulted phase (phase-a). These small fluctuations are due to the so-called low frequency electromechanical transients. It should be noted that the ideal source model cannot produce such phenomenon. With reference to the time plots of the load angle change and the rotor speed deviations (figs. 8.52 and 8.53), it can be seen that the system stability can still be maintained even with the faulted phase being completely disconnected from the system for any length of time.

– **Phase-a to earth, near voltage zero fault:**

Figs. 8.54 and 8.55 represent the voltages and currents obtained using the two aforementioned source models for a midpoint phase-a near voltage zero fault. It can be seen that the dc current offsets are very prominent for a fault which occurs near voltage zero. This conforms to theory because the dc transient exponential component term reaches a maximum when the fault instant is at voltage zero, and reaches a minimum when the fault instant is near voltage maximum. With reference to fig. 8.55, it can be seen that two models produce different dc current offsets, and hence different current zero crossing times. This is because the reactance to resistance ratio ($X : R$ ratio) of the actual generator model is higher than that of the ideal source model. It should be noted that the higher the ratio, the larger will be the decaying time constant for the dc transient exponential component. Figs. 8.56 and 8.57 show the extended simulation of the sound phase currents (b and c) with a longer observation time. With reference to the above two figures, it can be seen that much bigger fluctuations of sound phase currents (phase-b and c) are observed after the pole opening of the faulted phase (phase-a). With reference to figs. 8.58 and 8.59, it can be seen that for the near voltage zero fault, the load angle change and the rotor speed deviations are larger than those observed for the near voltage maximum fault (fig. 8.52 and 8.53). However, the system stability can still be maintained with the faulted phase being disconnected for any length of time.

- **Three-phase to earth fault:**

  Figs. 8.60 and 8.61 show the sending end voltages and currents comparison between the two aforementioned models for a mid-point three phase to earth fault. With reference to fig. 8.61, it can be seen that the dc current offsets between the two models are different for all the three currents, and the zero crossings of the fault currents are different for the two models concerned.

- **Simulation of full autoreclosure sequence:**

  Figs. 8.62 and 8.63 show the three phase autoreclosure sequence for a mid-point three phase to earth fault. Figs. 8.64 and 8.65 show the single pole autoreclosure sequence for a mid-point phase-a to earth fault. Similar observations, as described in section 8.4 for the simulation of full autoreclosure sequence, can also be found in the figures. However, two distinct points must be made. Firstly, the low frequency electromechanical transients can be clearly observed in the sound phase currents of the single pole sequence during the so-called 'dead period'. Secondly, since the breaker poles are opened at the corresponding current zeros of the respective phases, there is no current chopping and the transient distortions for the voltages during the breaker pole opening stage are significantly reduced.

- **Discussion:**

  The developments in generator design and the associated trends in machine parameters, have resulted in using higher reactance to resistance ratio ($X : R$ ratio) appearing at the high voltage terminals of the generator transformer. This increase, coupled with the development

to reduce the copper losses and the use of multiple conductor transmission lines, has resulted in a general raising of the network $X : R$ level. This indicates that under the worst fault conditions, very slow decaying of the dc current will be produced and very high fault current asymmetry can occur [42].

Results obtained have shown that the simplified source model does not produce the same degree of dc offset as that produced by the actual generator model. It has been found that for faults which occur near zero voltage, complete offsetting of the current waveforms can occur and significant differences are observed when compared to those obtained using simplified source models. For faults which occur near the peak of the prefault voltage, the waveforms do not differ significantly from those observed using simplified source models based upon subtransient values. Furthermore, it should be mentioned that the simplified source model cannot produce the low frequency electromechanical transient phenomenon, which is clearly shown when the actual generator model is used.

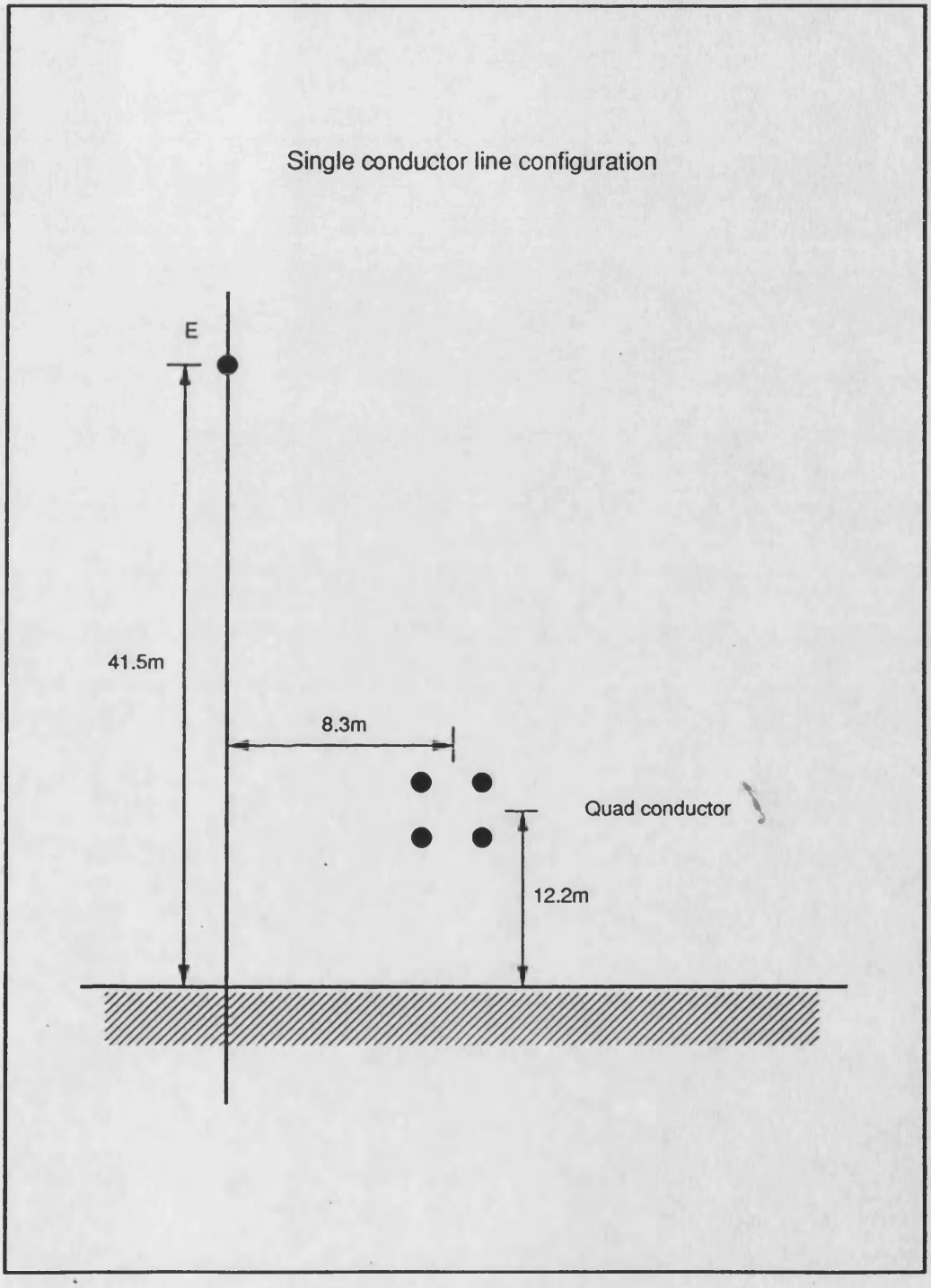

Figure 8.1: Single conductor transmission line configuration

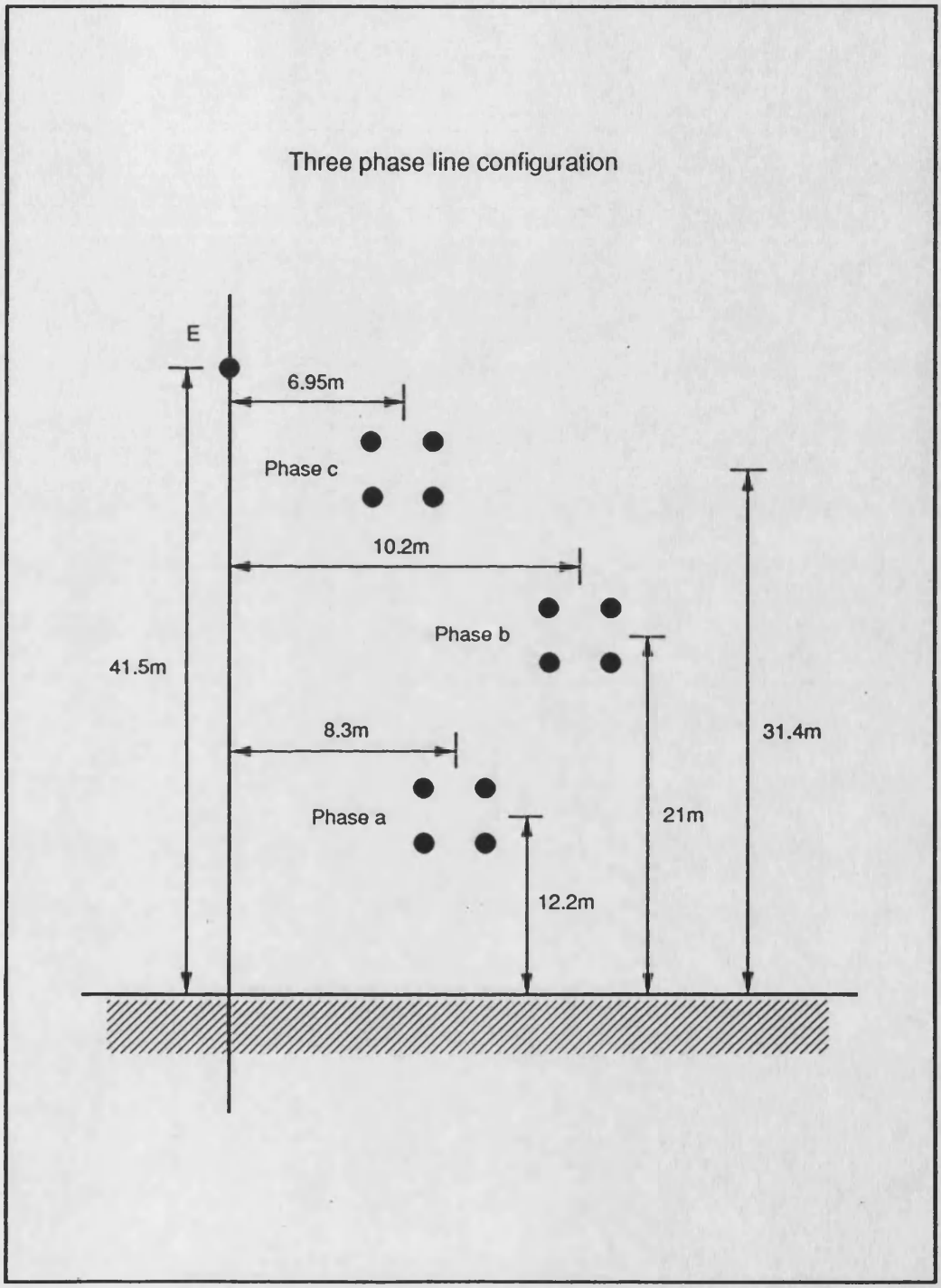

Figure 8.2: Three phase transmission line configuration

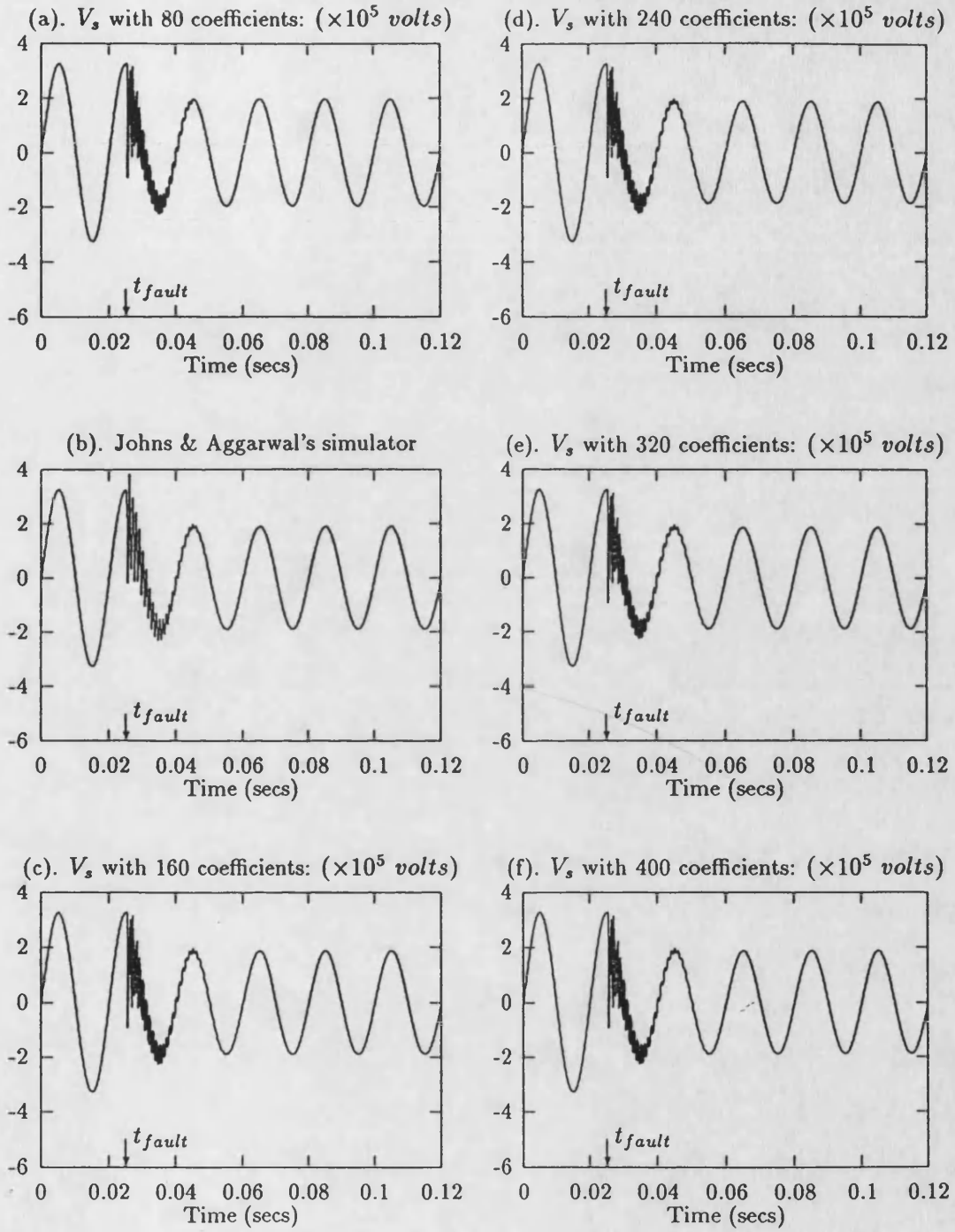

Figure 8.3: Single conductor modelling: sending end voltage. (a), (c)-(f): time convolutional technique. (b): mainframe simulator by Johns & Aggarwal.

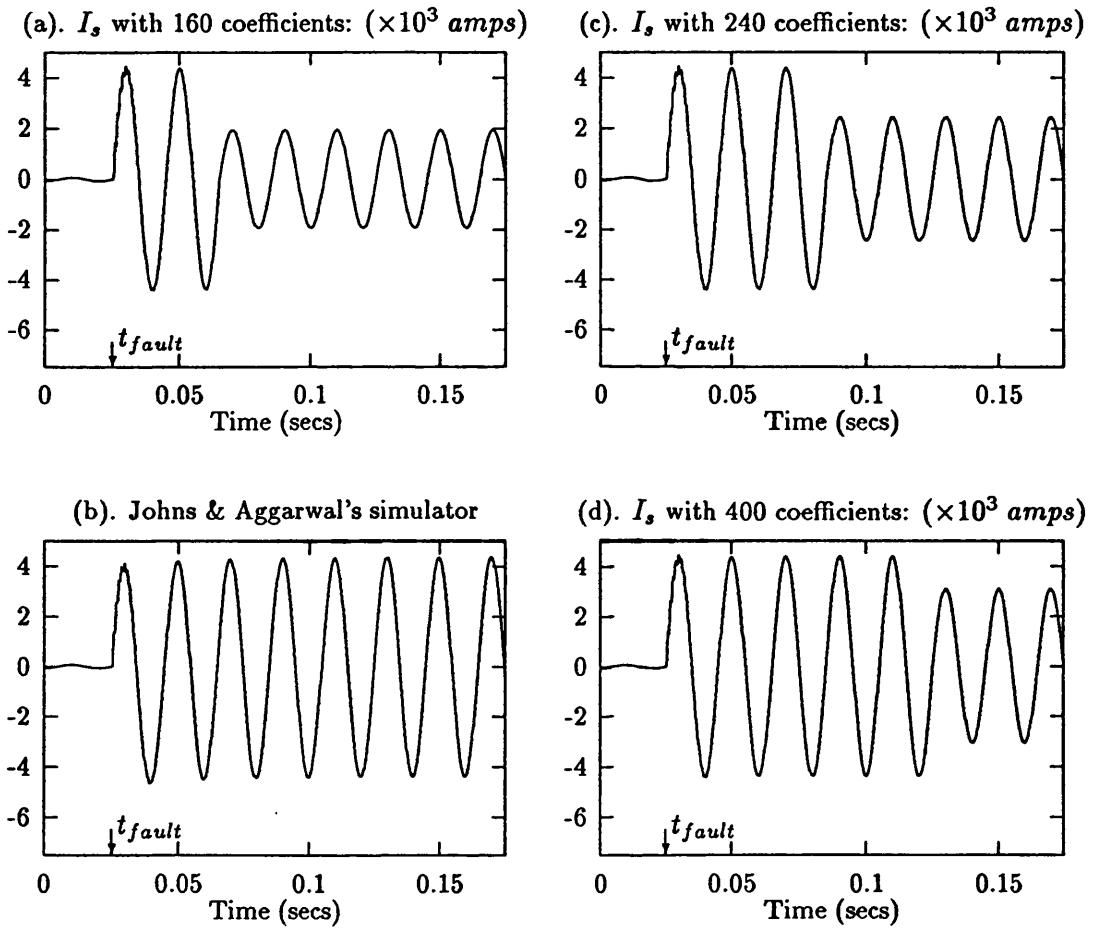

Figure 8.4: Single conductor modelling: sending end current. (a), (c) and (d): time convolutional technique. (b): mainframe simulator by Johns & Aggarwal.

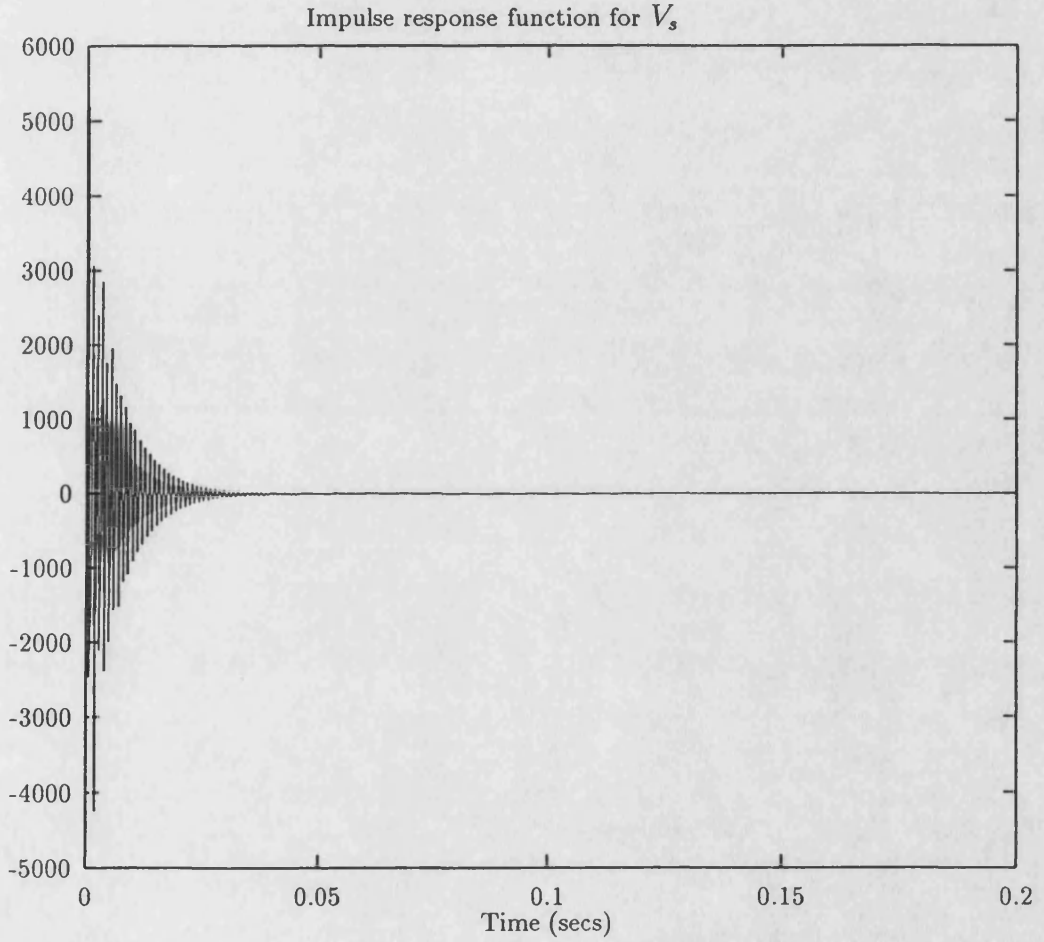

Figure 8.5: Impulse response function ($Z_{ss}Y_D$) for sending end voltage calculation.

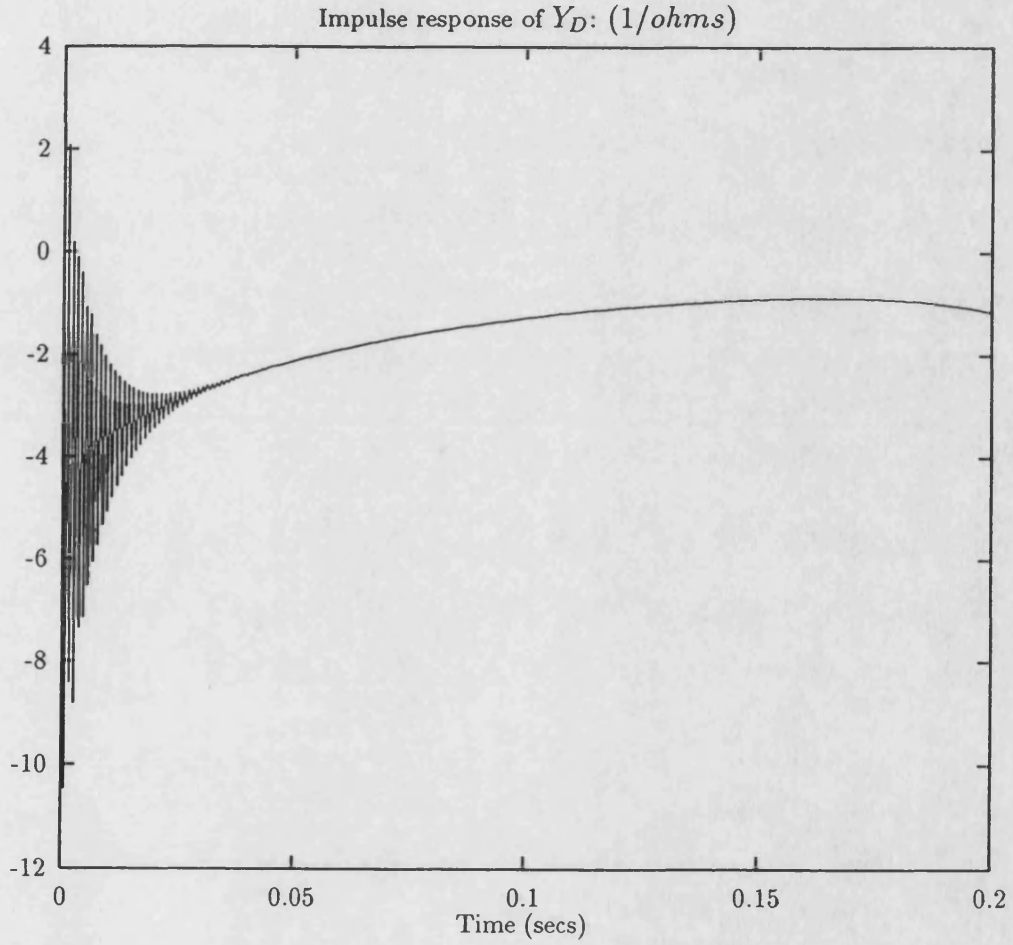

Figure 8.6: Impulse response function (Y_D) for sending end current calculation.

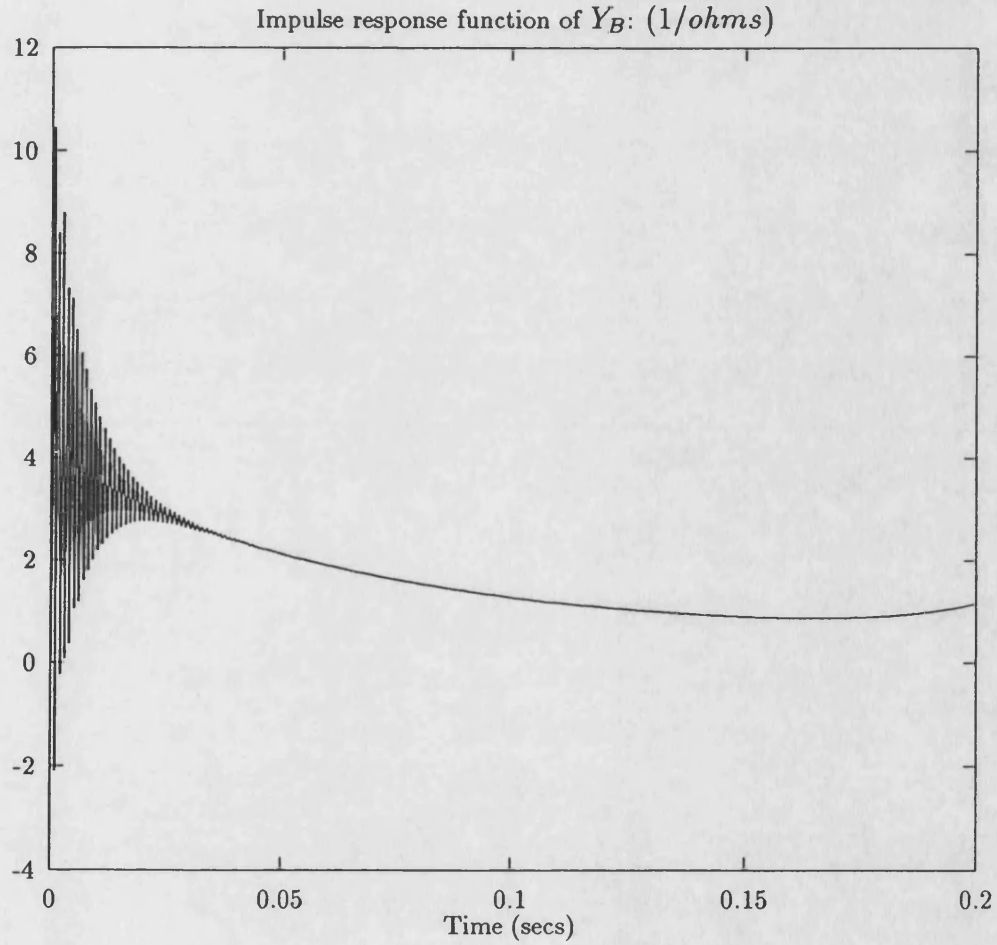

Figure 8.7: Impulse response function of Y_B.

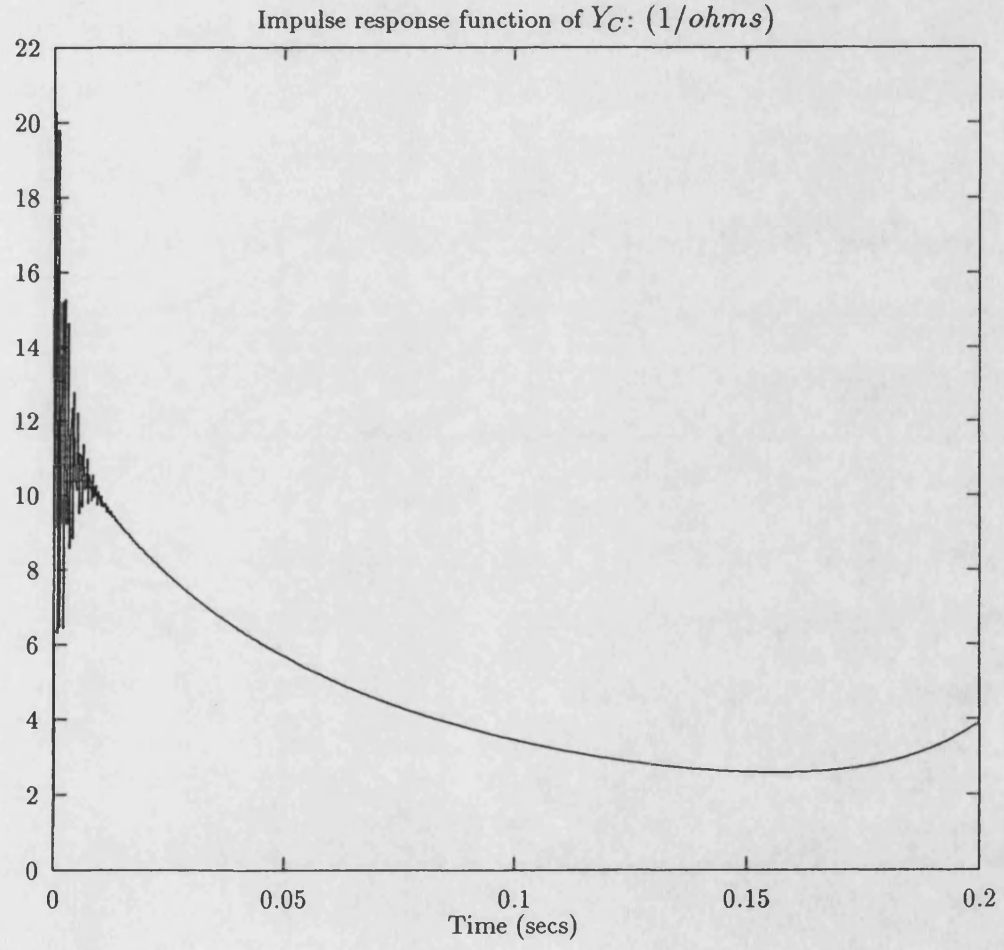

Figure 8.8: Impulse response function of Y_C.

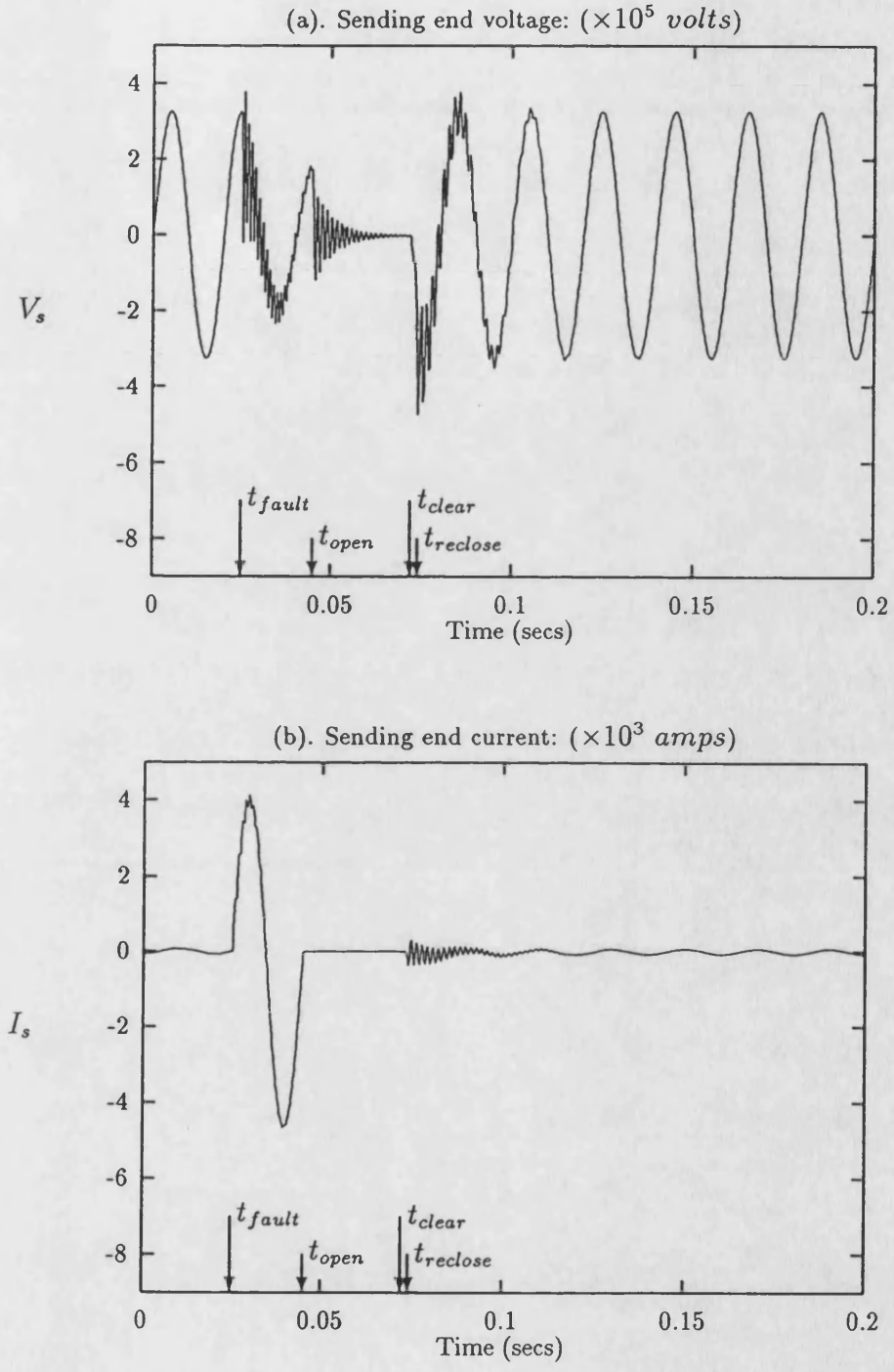

Figure 8.9: Frequency domain single conductor modelling: fault distance = 85km from sending end, sending *s.c.l.* = 5GVA, receiving *s.c.l.* = 35GVA.

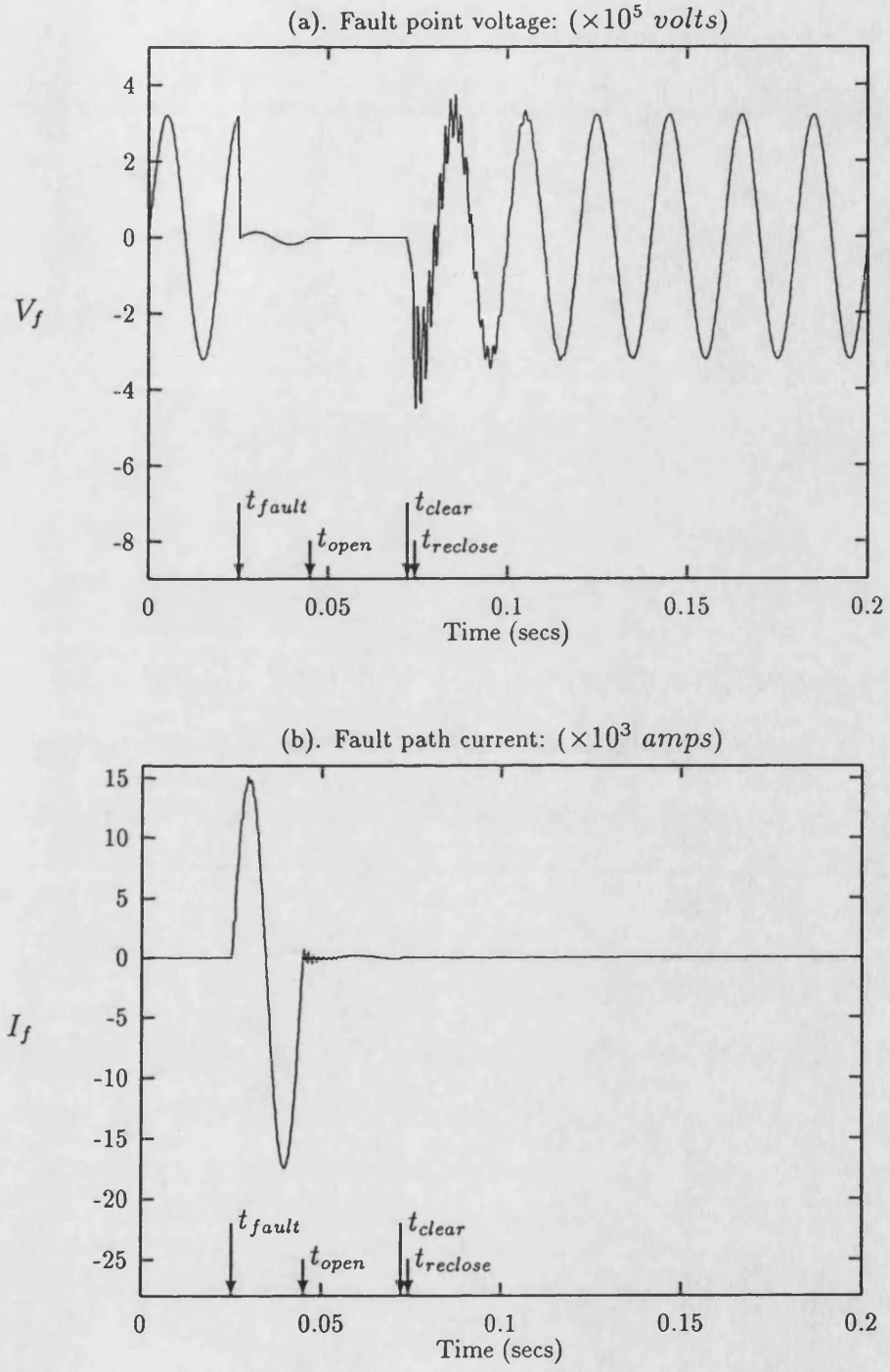

Figure 8.10: Frequency domain single conductor modelling: fault distance = 85km from sending end, sending *s.c.l.* = 5GVA, receiving *s.c.l.* = 35GVA.

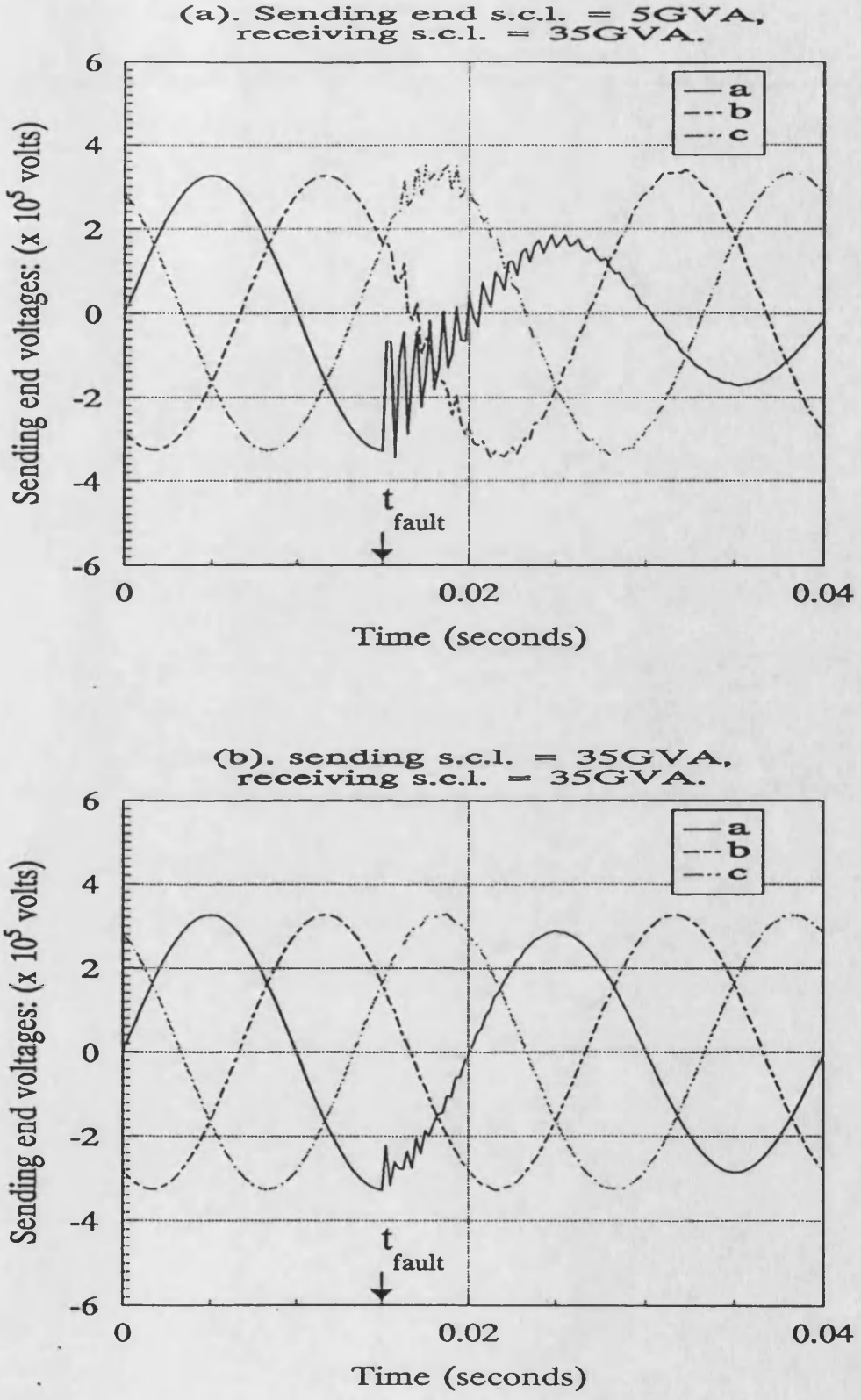

Figure 8.11: Frequency domain modelling: effect of source capacities on voltages, midpoint phase-a to earth fault.

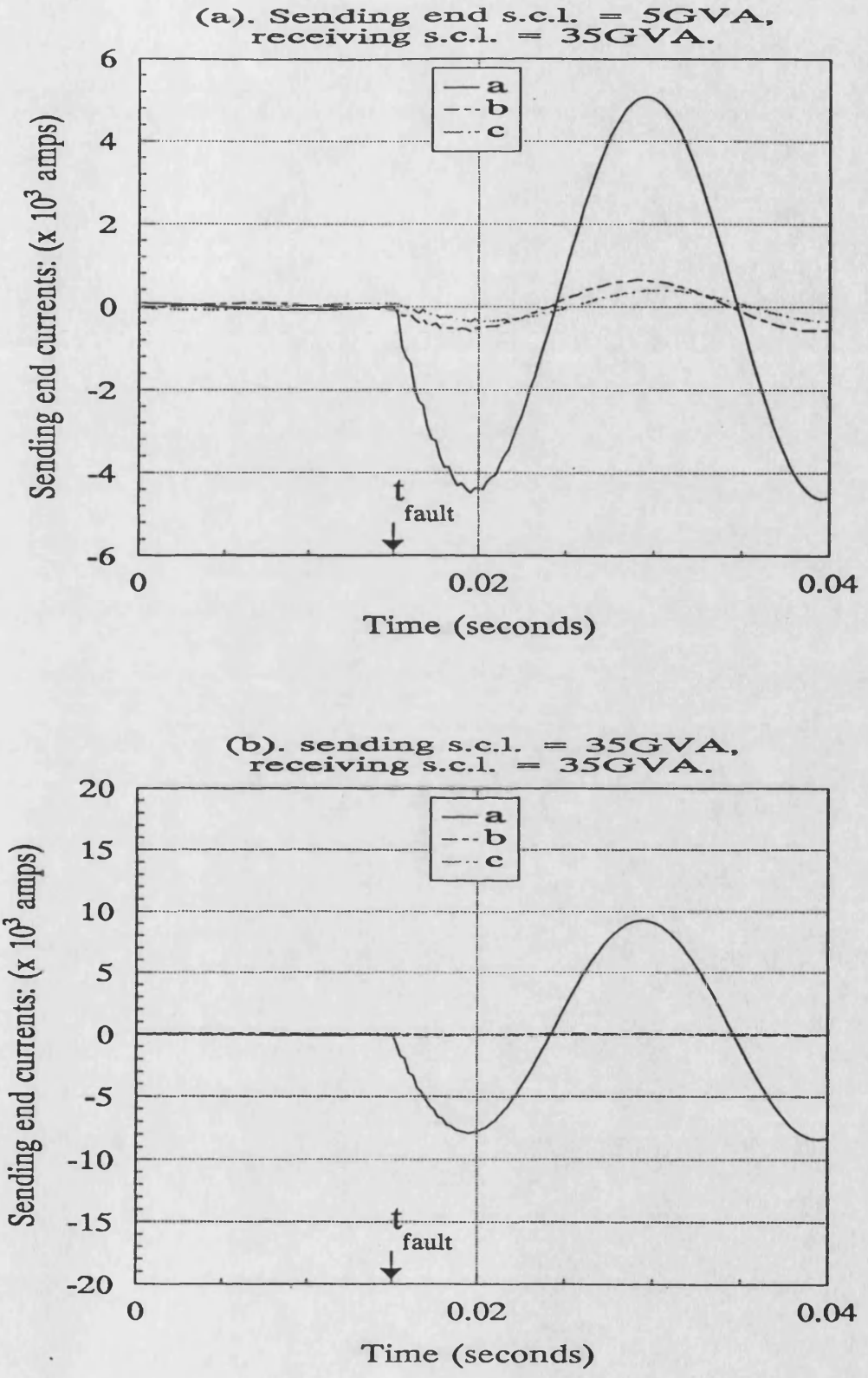

Figure 8.12: Frequency domain modelling: effect of source capacities on currents, midpoint phase-a to earth fault.

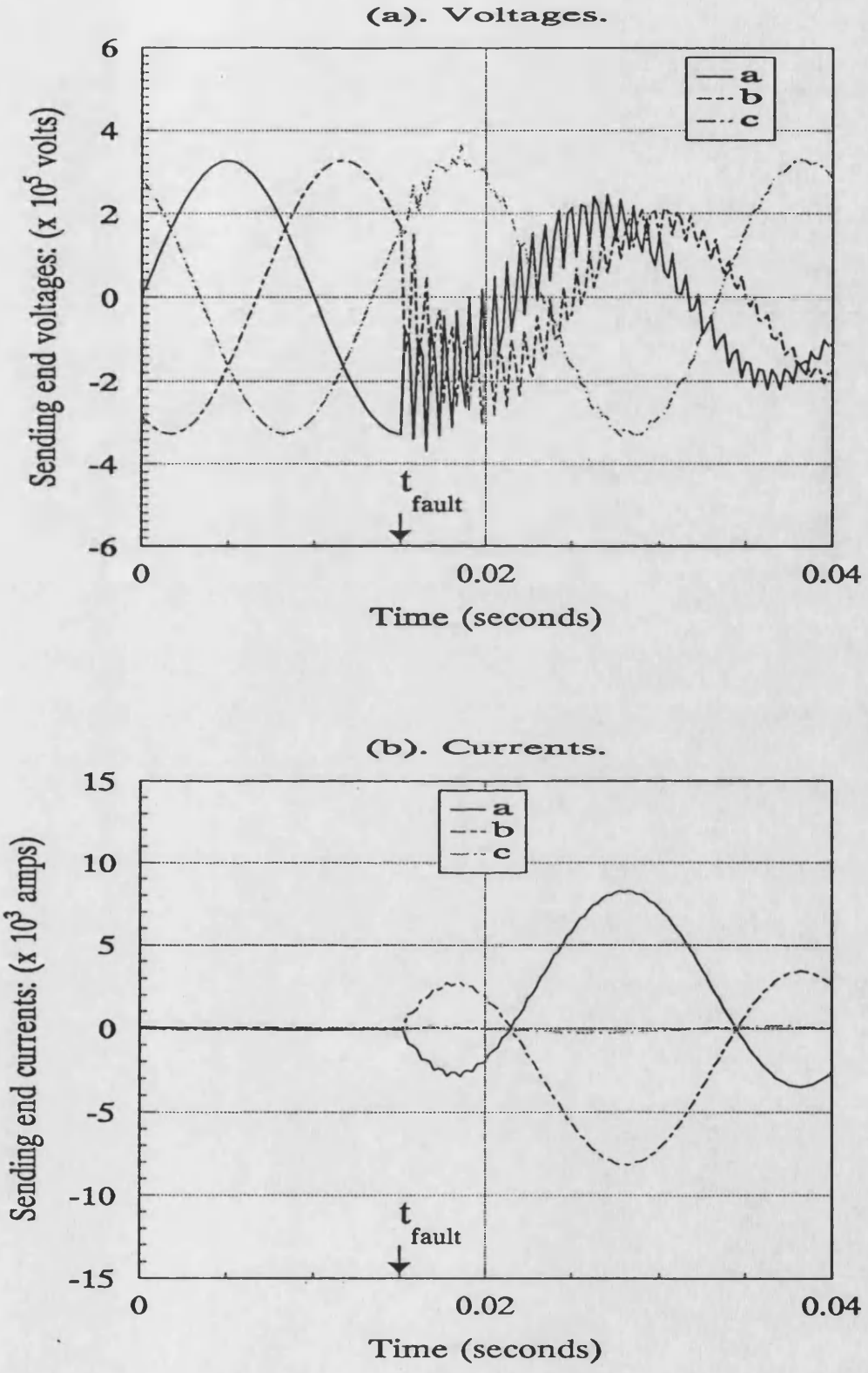

Figure 8.13: Frequency domain modelling: midpoint phase-ab fault, sending $s.c.l. = 5$GVA, and receiving $s.c.l. = 35$GVA.

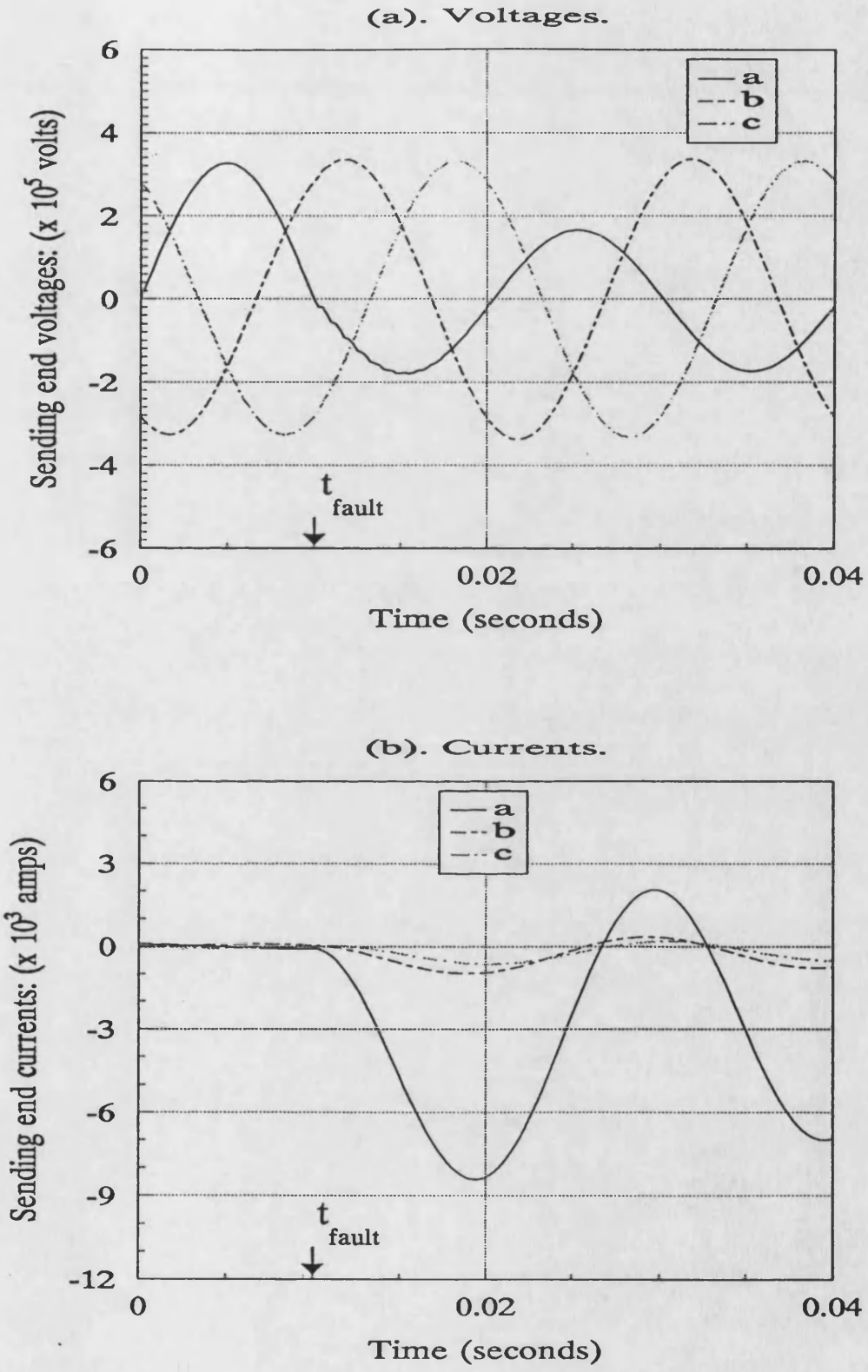

Figure 8.14: Frequency domain modelling: a-earth fault at near voltage zero, sending $s.c.l. = 5GVA$ and receiving $s.c.l. = 35$GVA

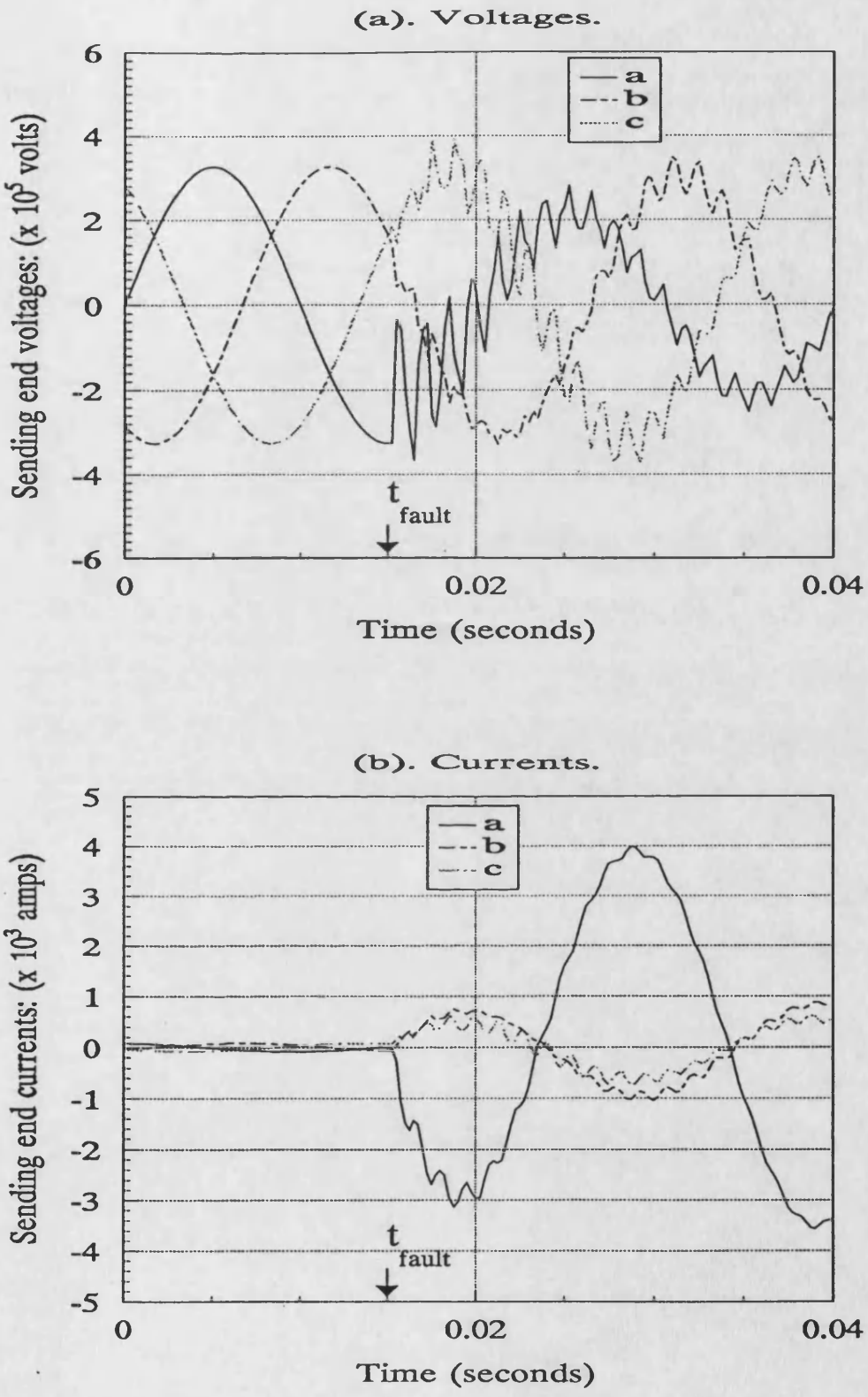

Figure 8.15: Frequency domain modelling: a-earth fault at receiving end, sending $s.c.l. = 5GVA$ and receiving $s.c.l. = 35$GVA

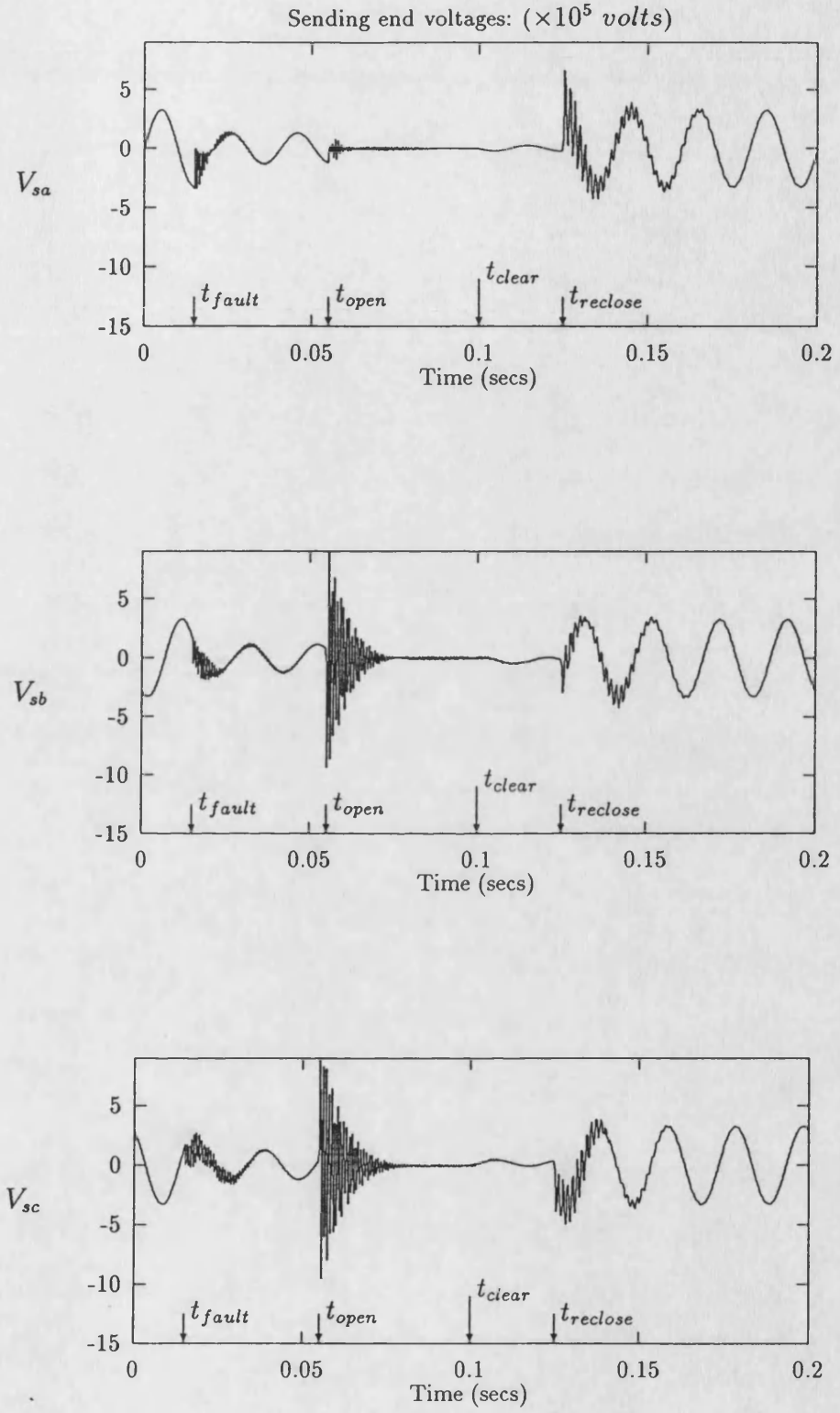

Figure 8.16: Frequency domain modelling: midpoint three-phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

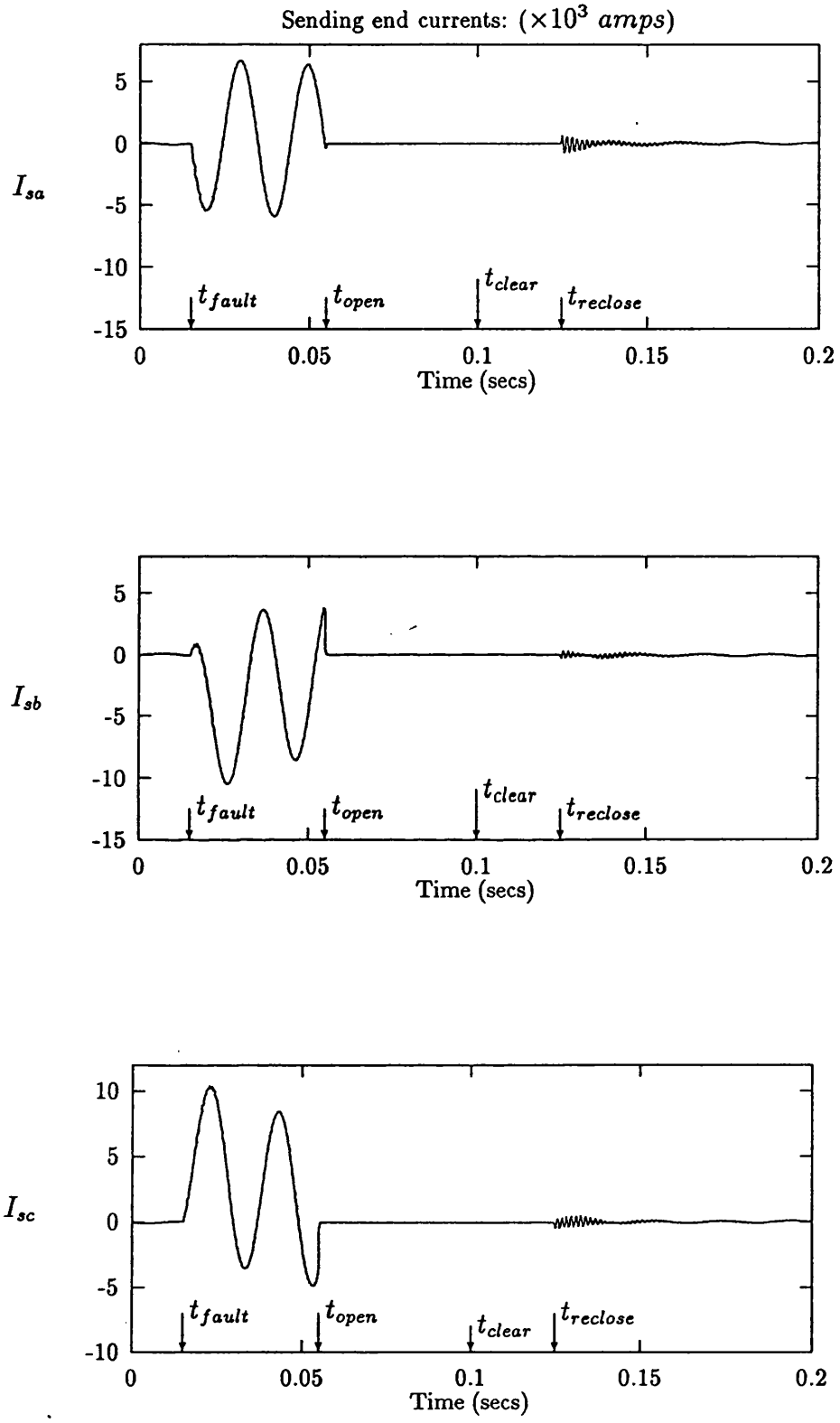

Figure 8.17: Frequency domain modelling: midpoint three-phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

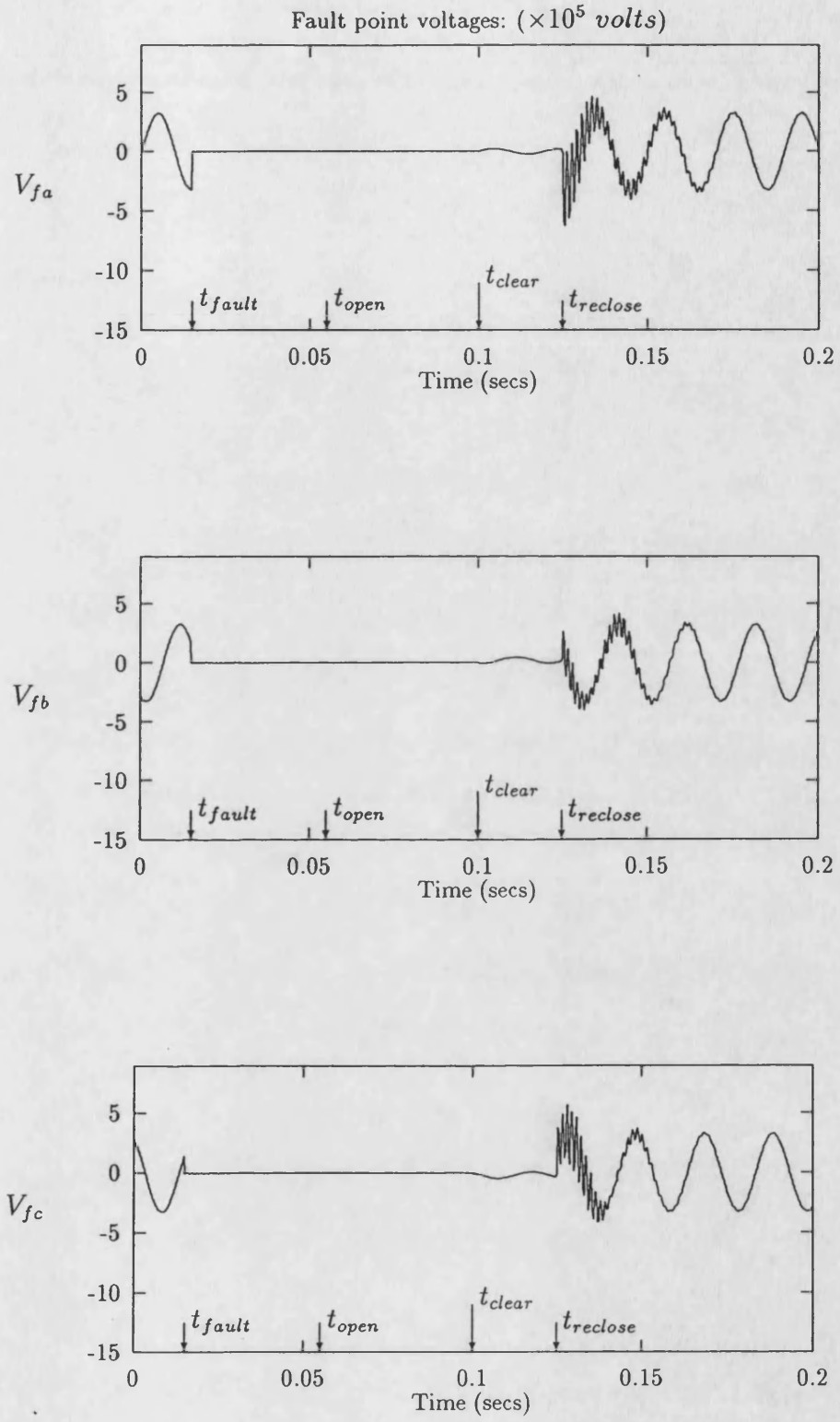

Figure 8.18: Frequency domain modelling: midpoint three-phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

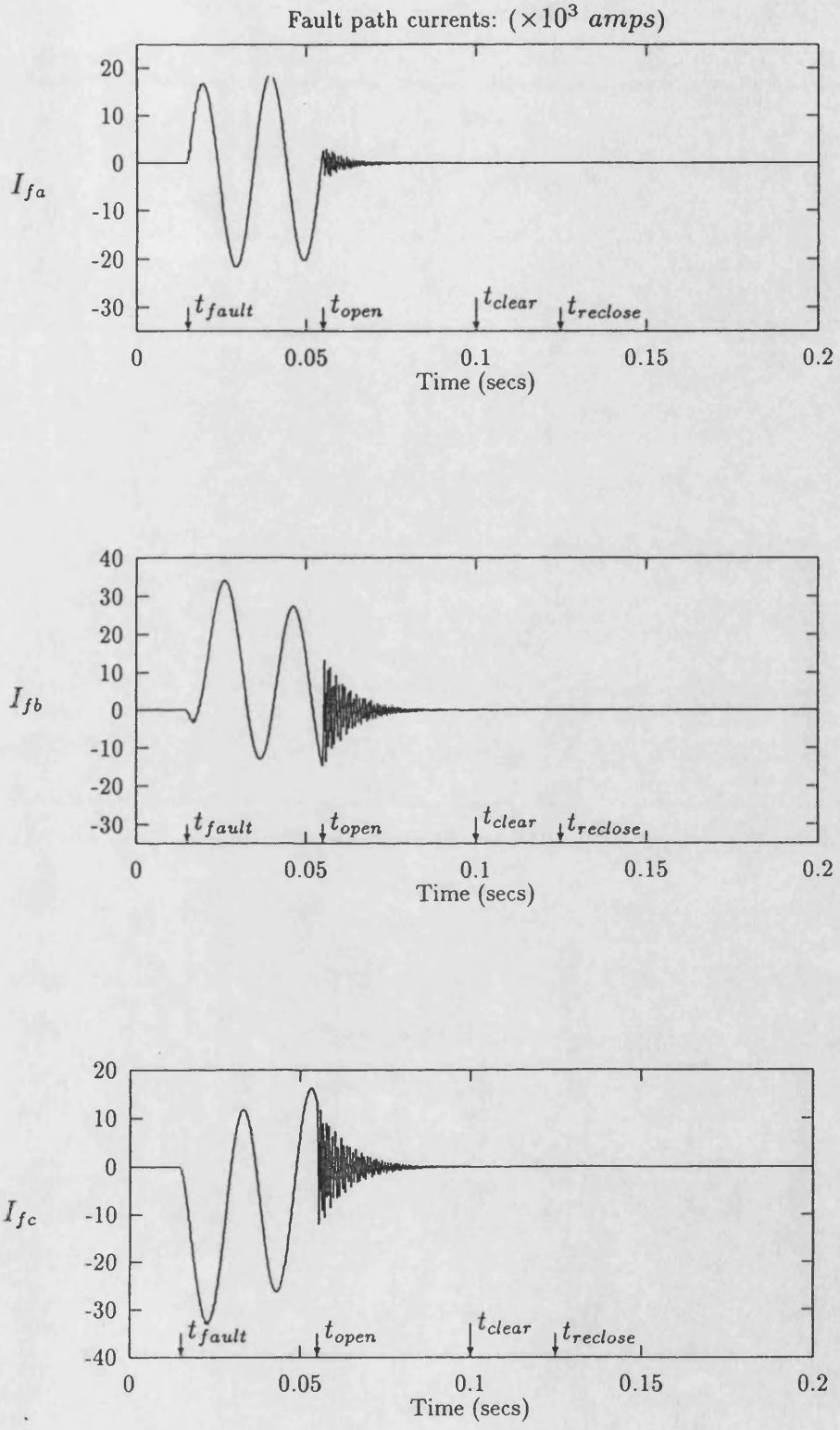

Figure 8.19: Frequency domain modelling: midpoint three-phase to earth fault, sending *s.c.l.* = 5GVA, receiving *s.c.l.* = 35GVA.

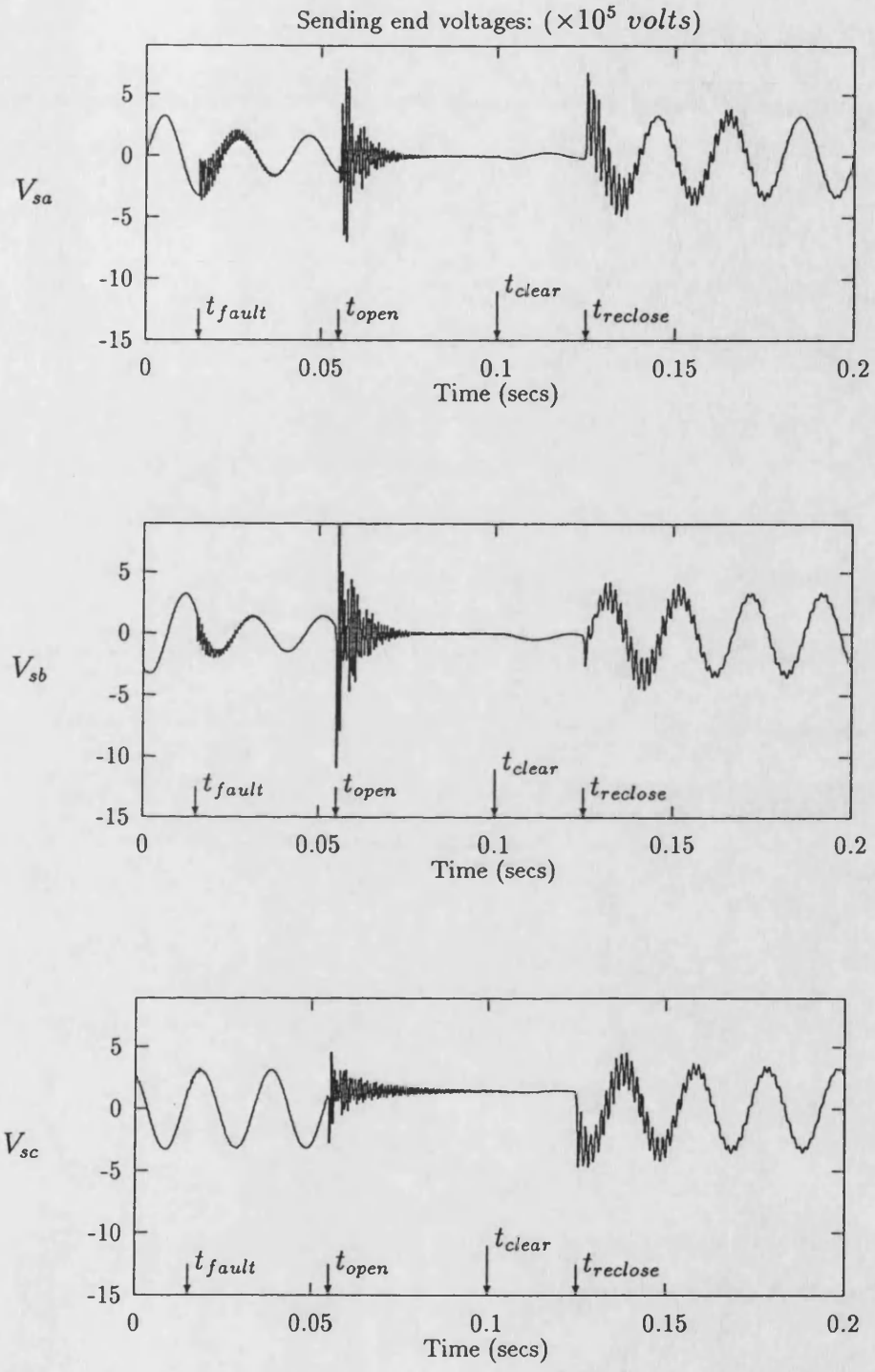

Figure 8.20: Frequency domain modelling: midpoint double phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

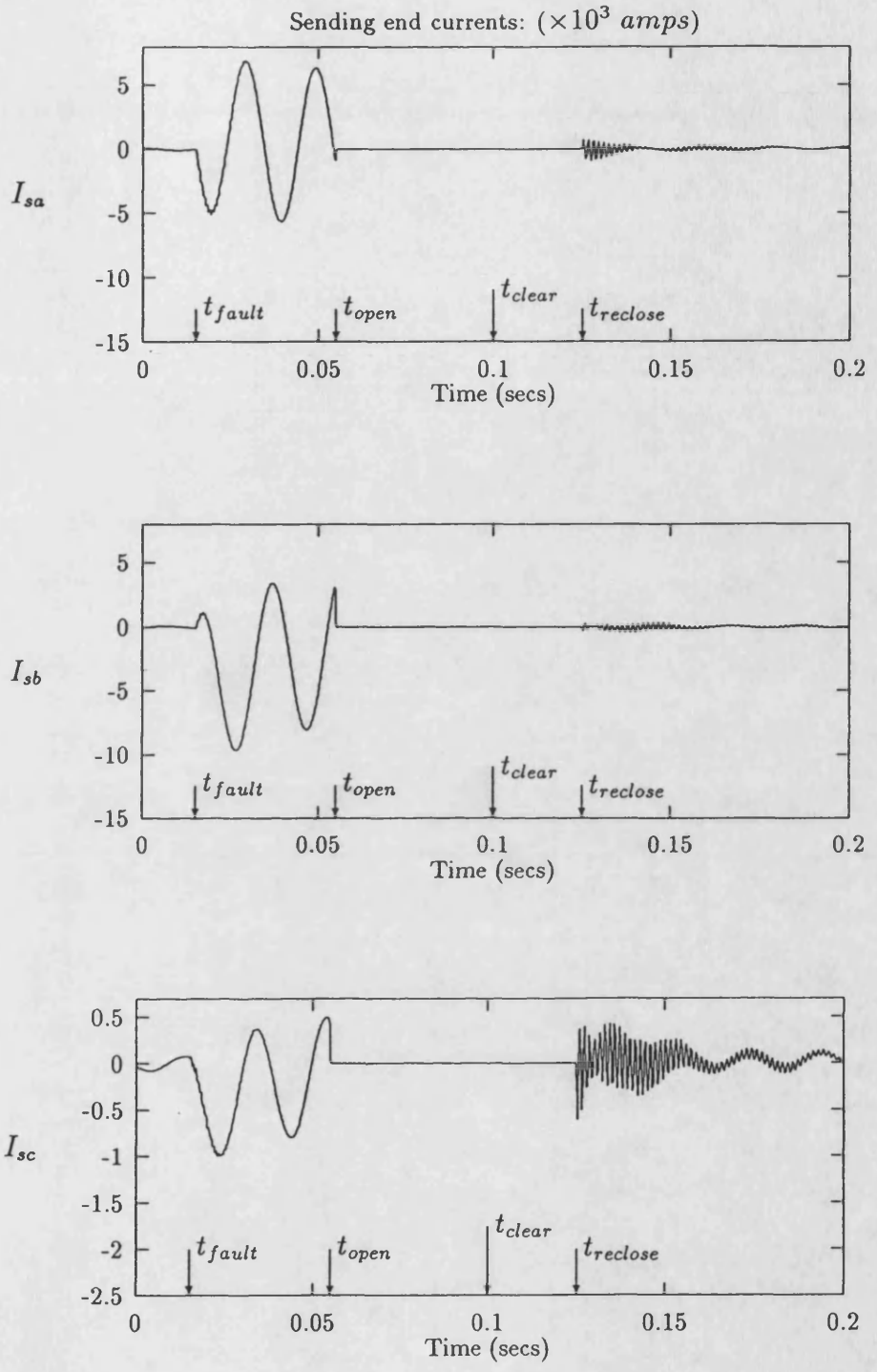

Figure 8.21: Frequency domain modelling: midpoint double phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

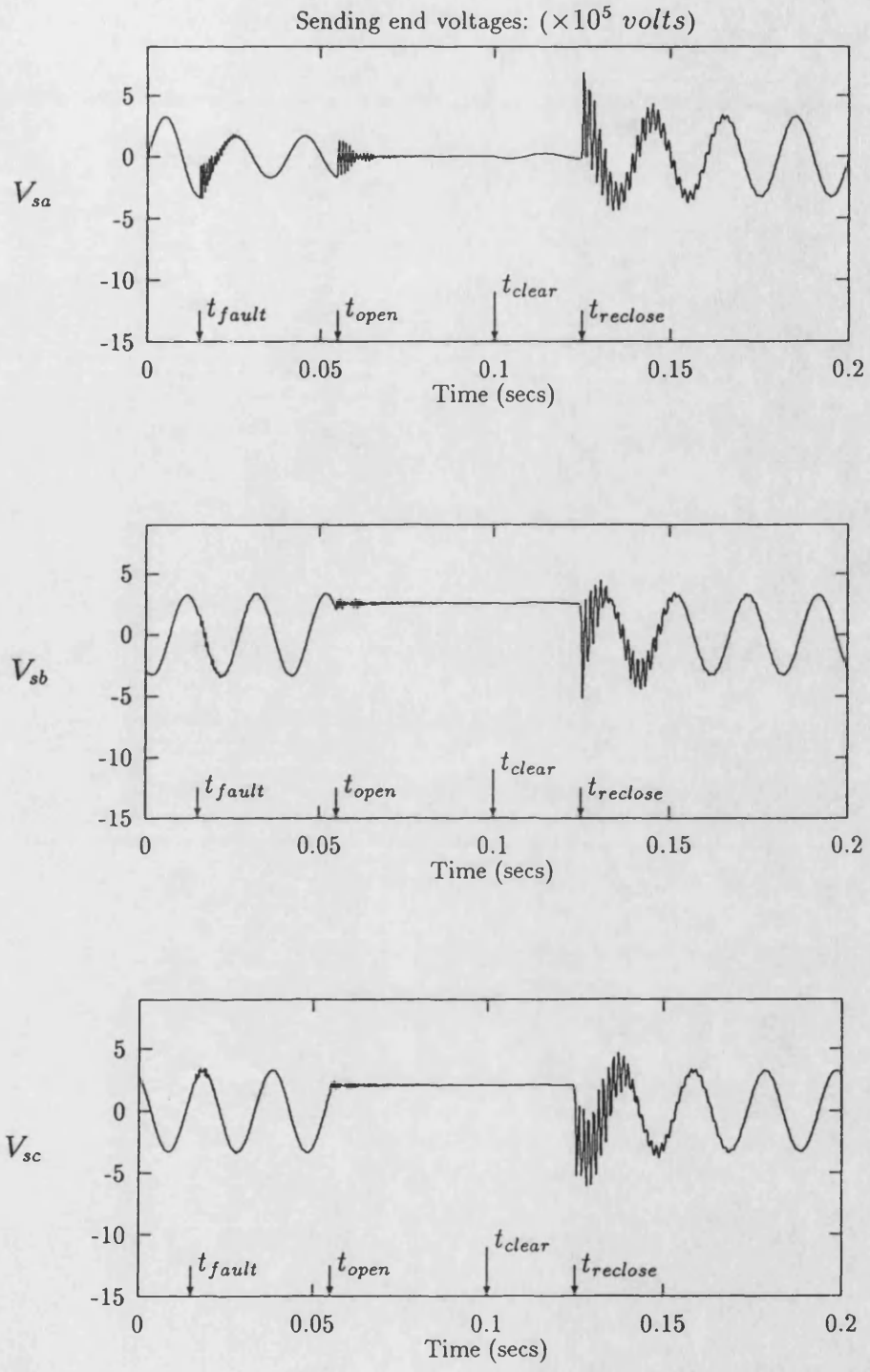

Figure 8.22: Frequency domain modelling: midpoint a-phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

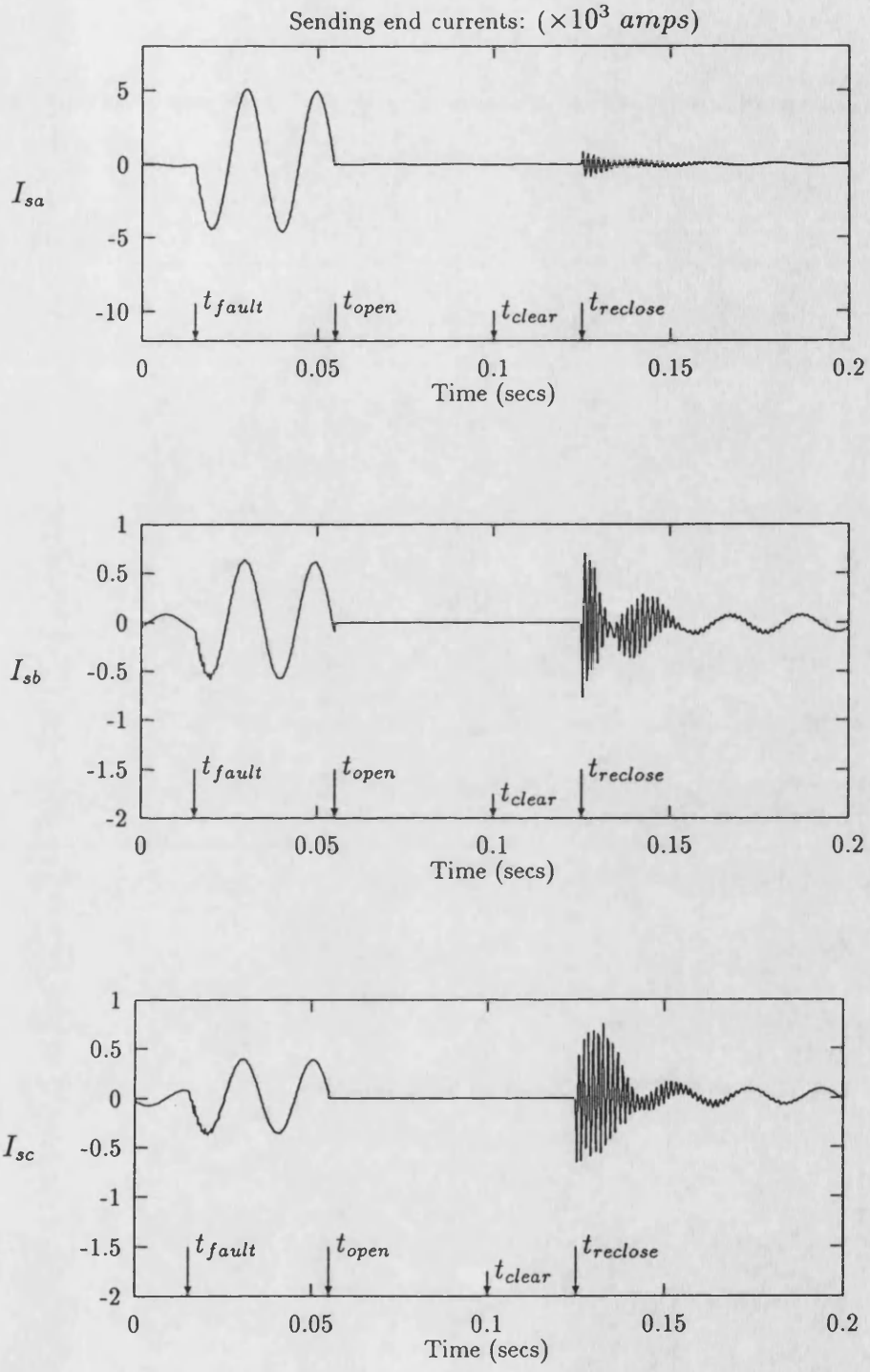

Figure 8.23: Frequency domain modelling: midpoint a-phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

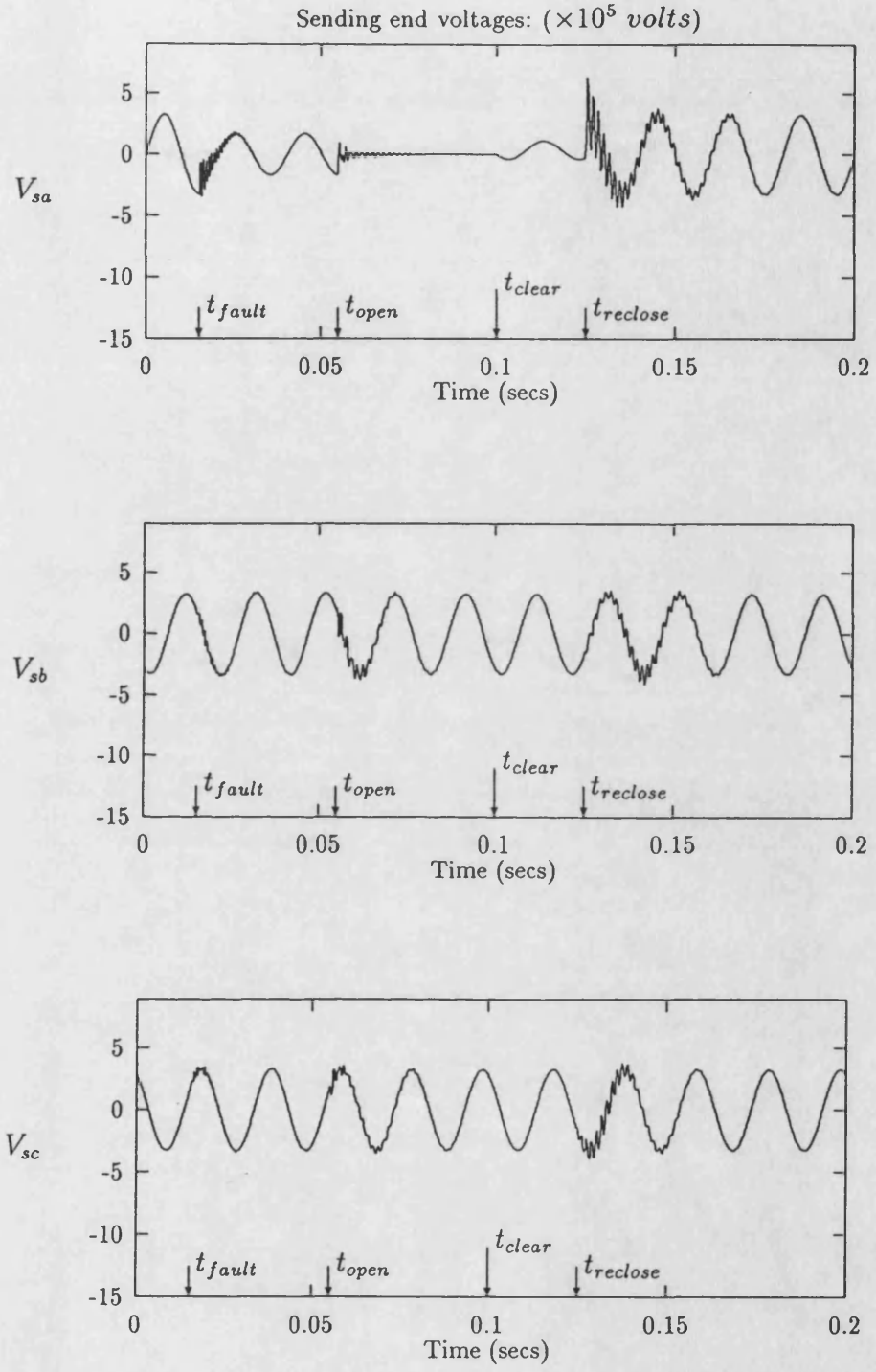

Figure 8.24: Frequency domain modelling: midpoint a-phase to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 35GVA.

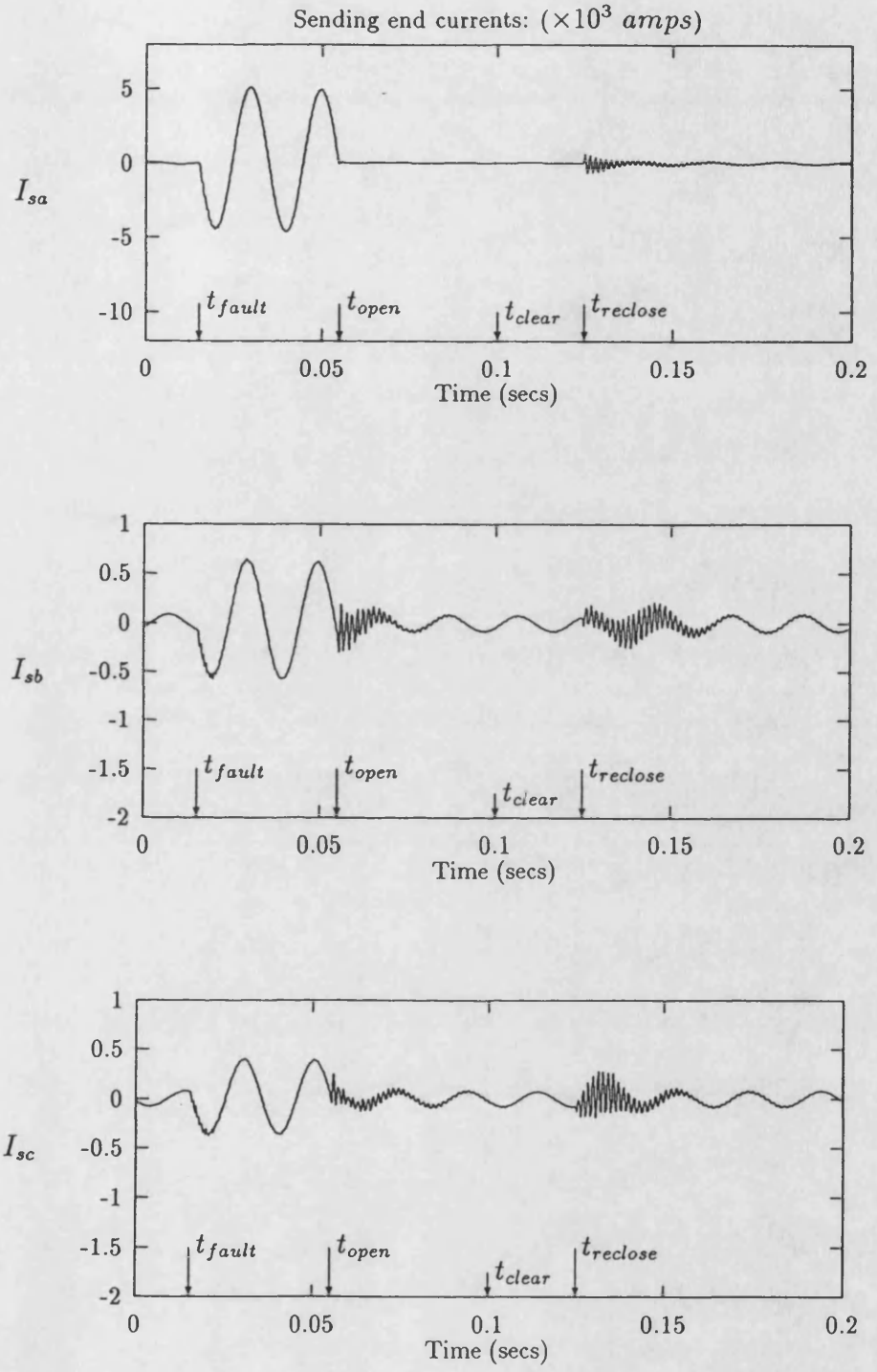

Figure 8.25: Frequency domain modelling: midpoint a-phase to earth fault, sending $s.c.l. = 5$GVA, receiving $s.c.l. = 35$GVA.

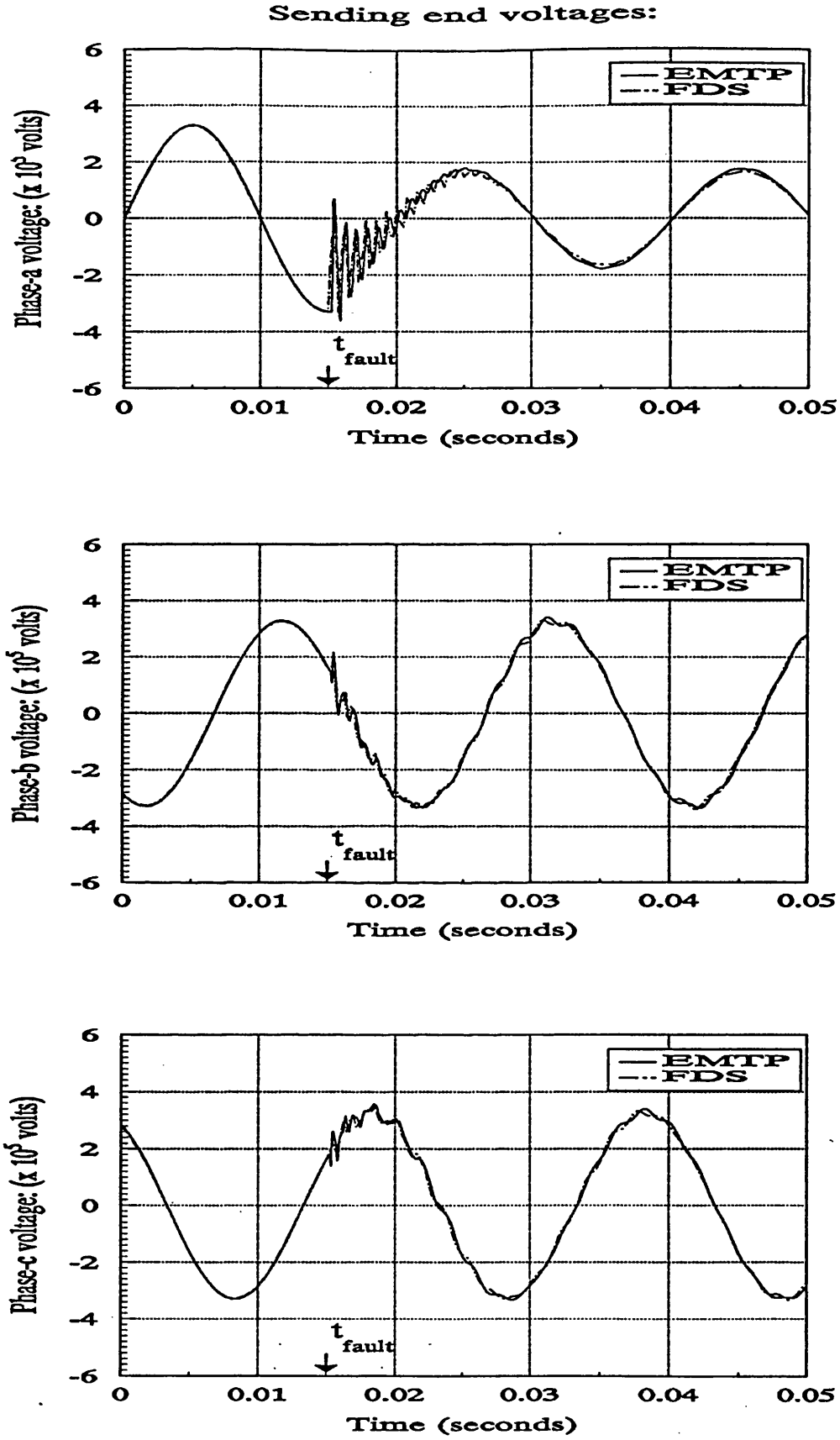

Figure 8.26: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

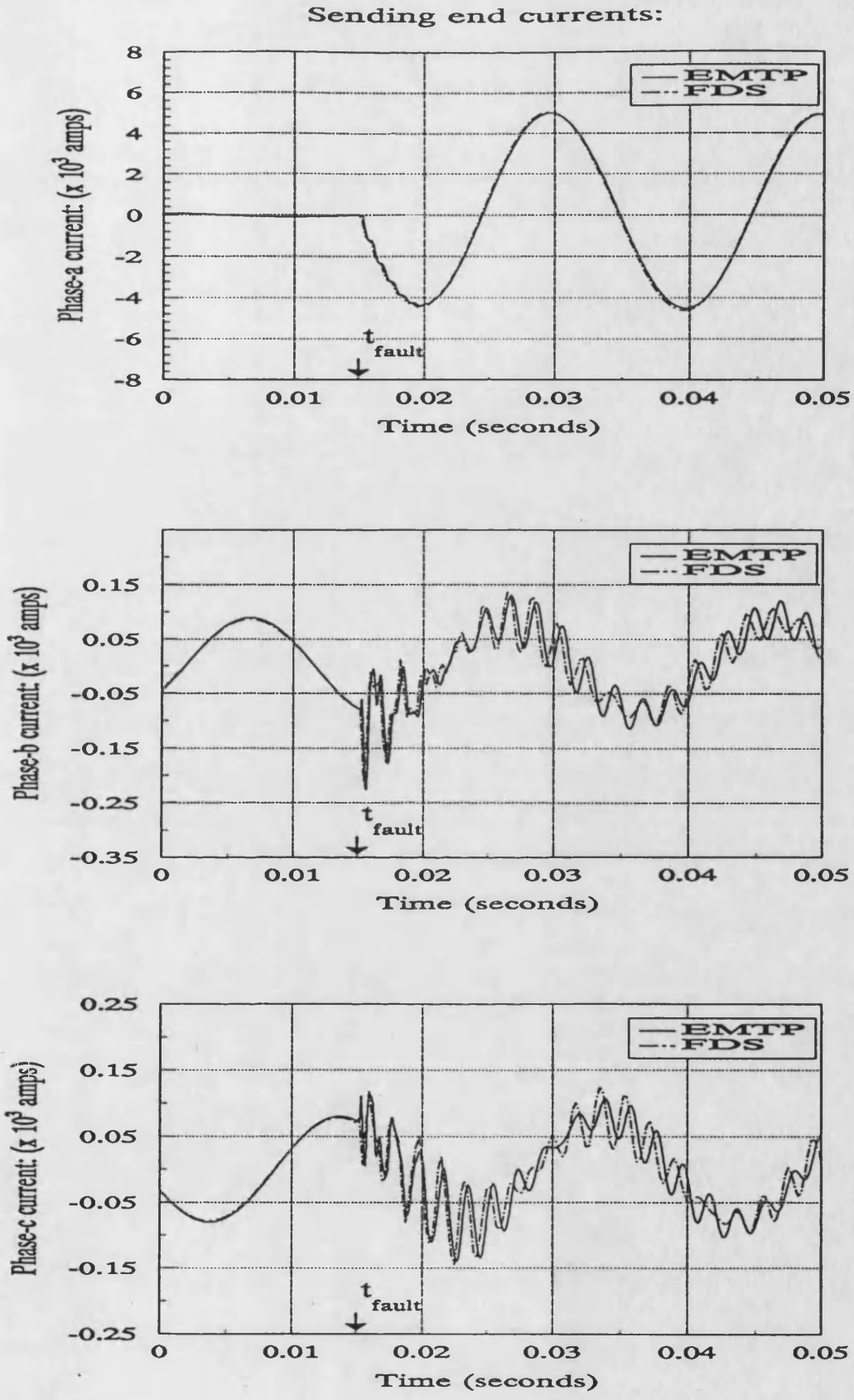

Figure 8.27: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

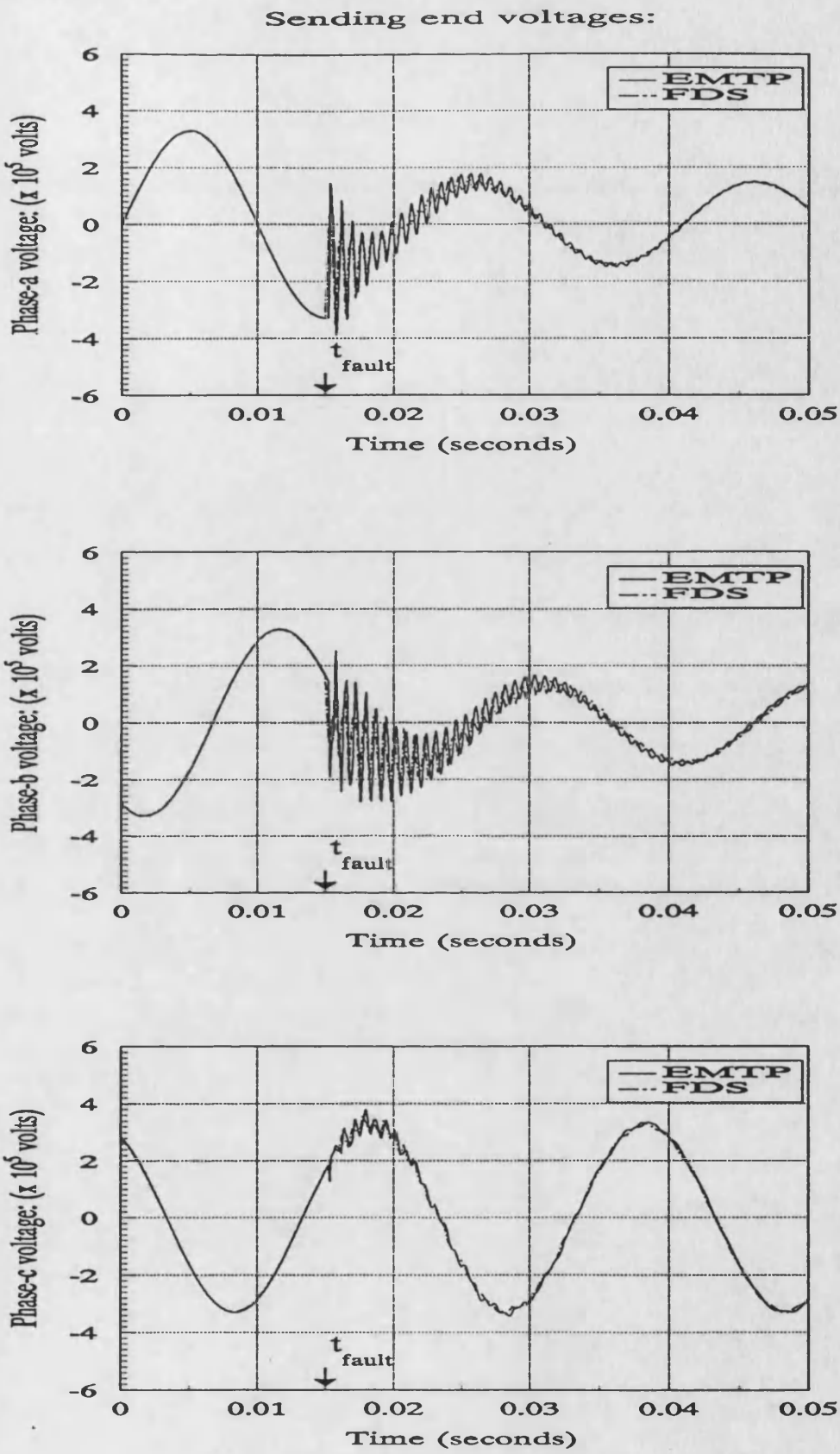

Figure 8.28: EMTP and FDS comparison: midpoint phase-a to earth fault, sending *s.c.l.* = 5GVA, receiving *s.c.l.* = 5GVA.

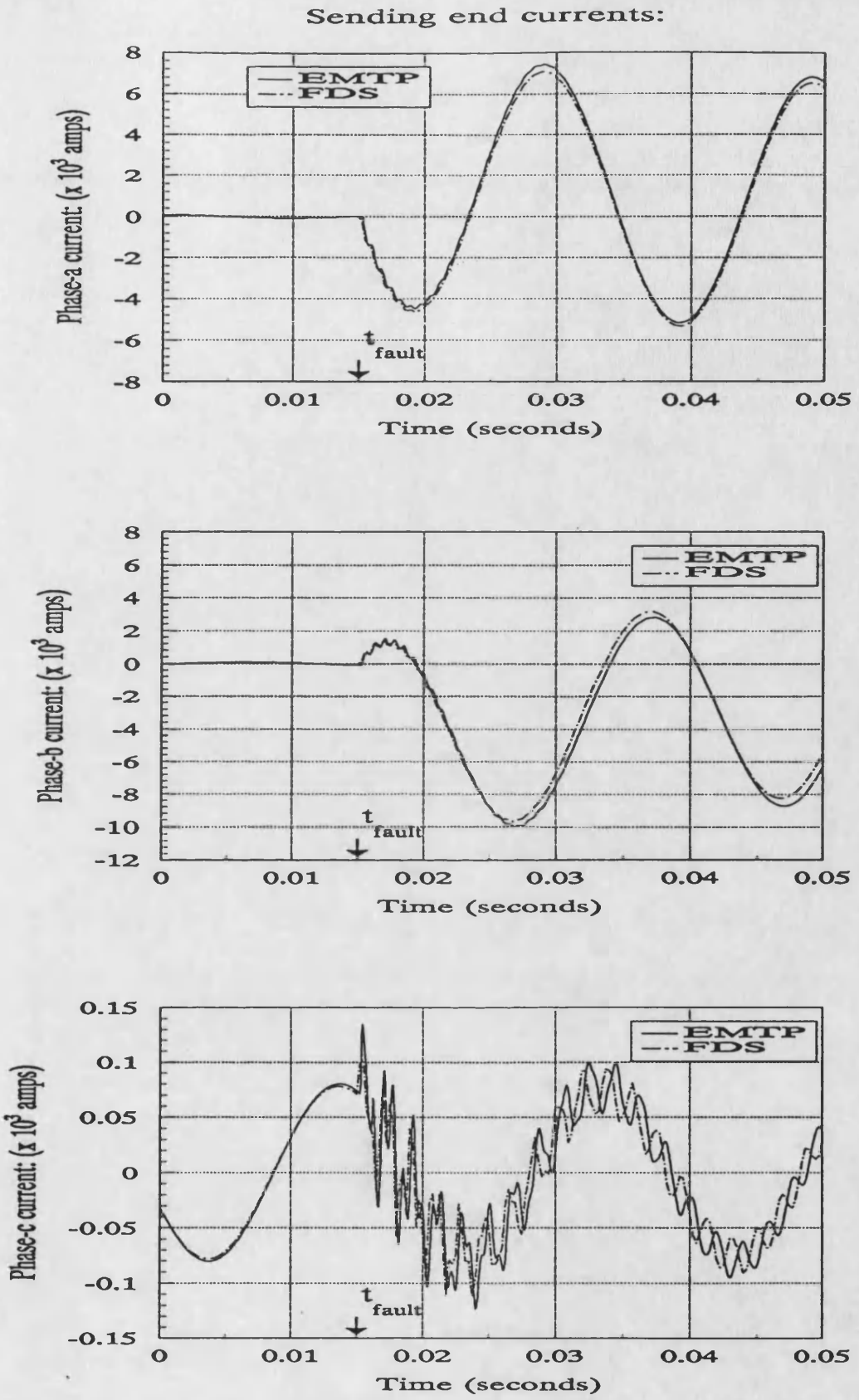

Figure 8.29: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

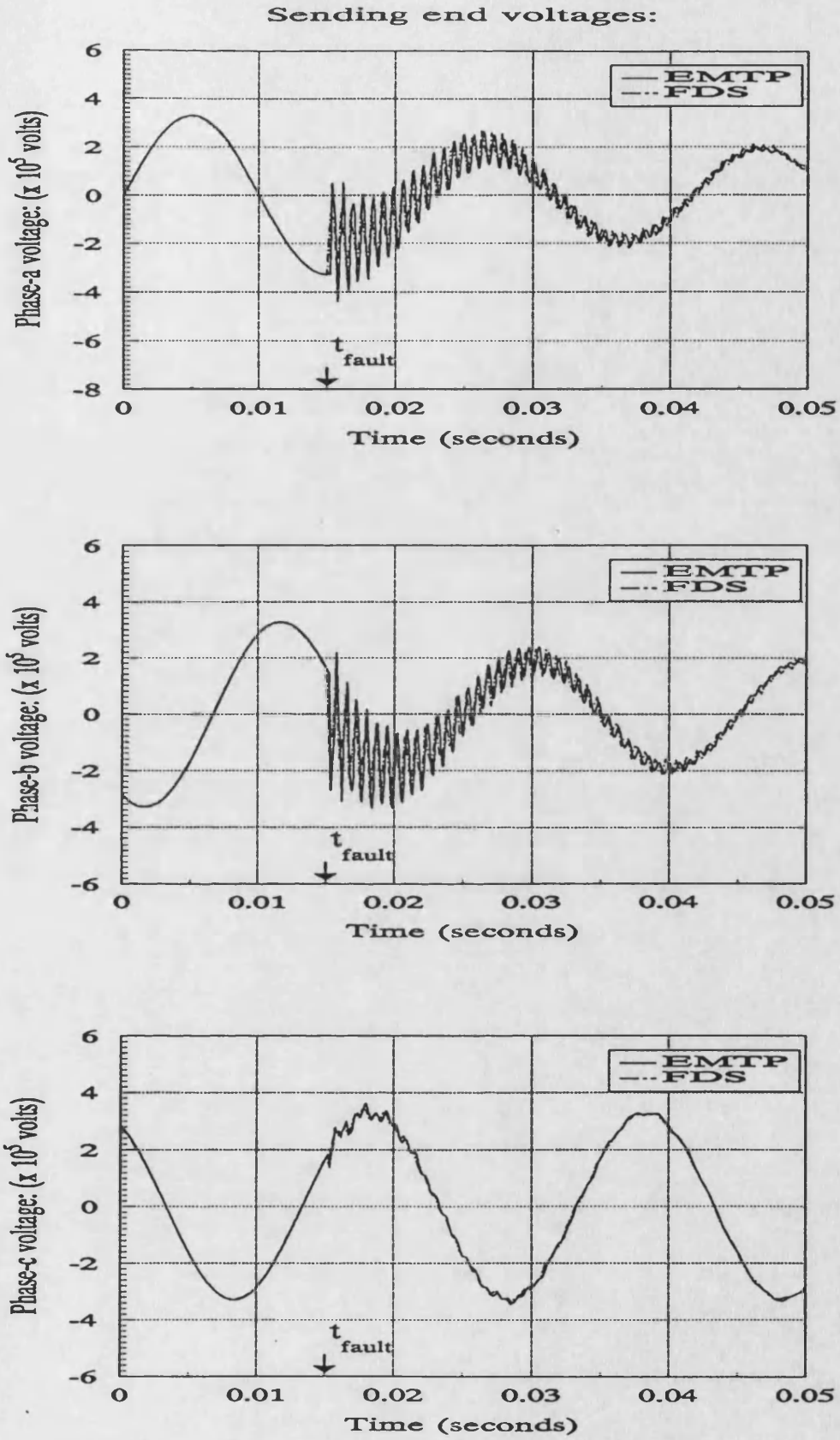

Figure 8.30: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

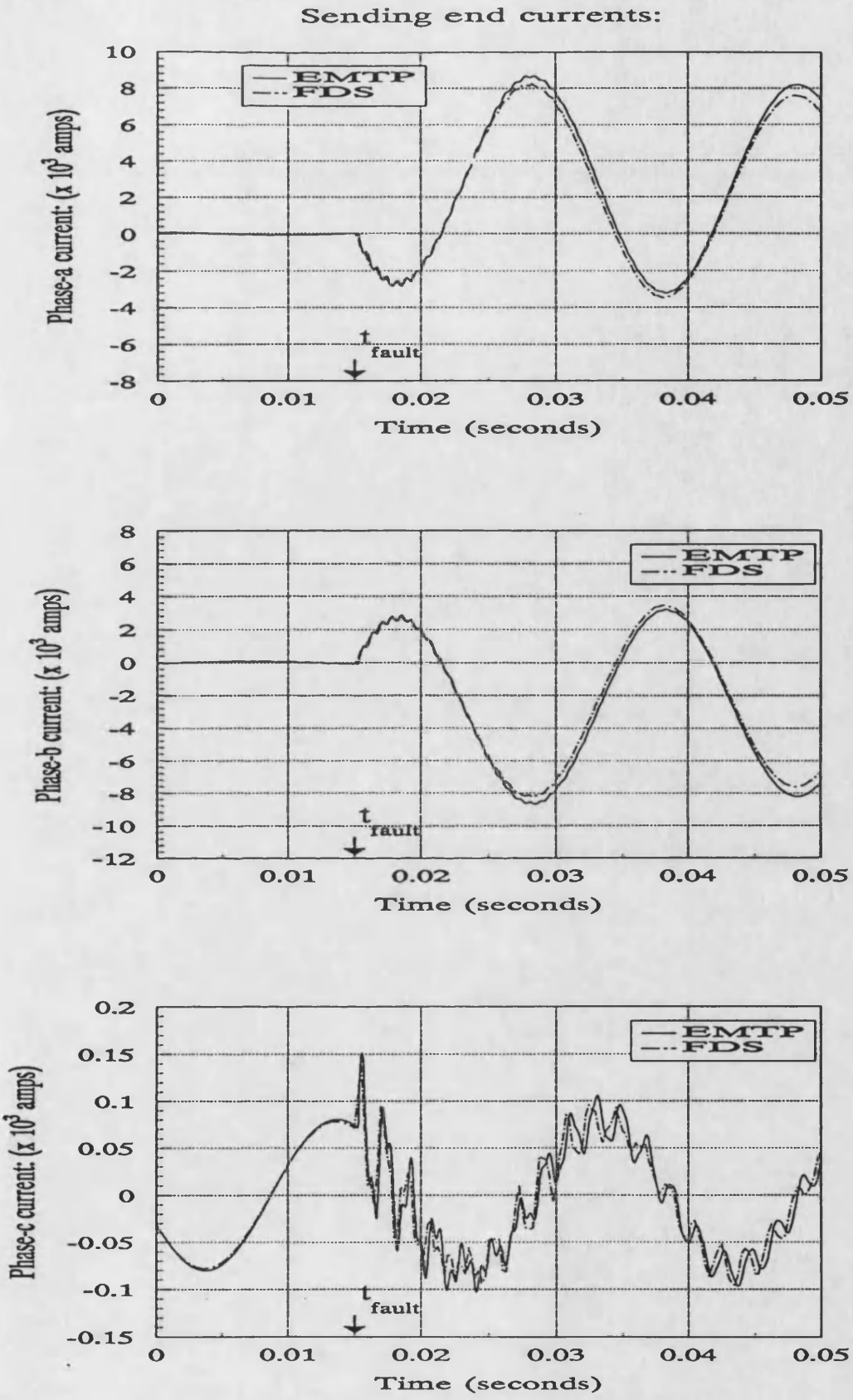

Figure 8.31: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

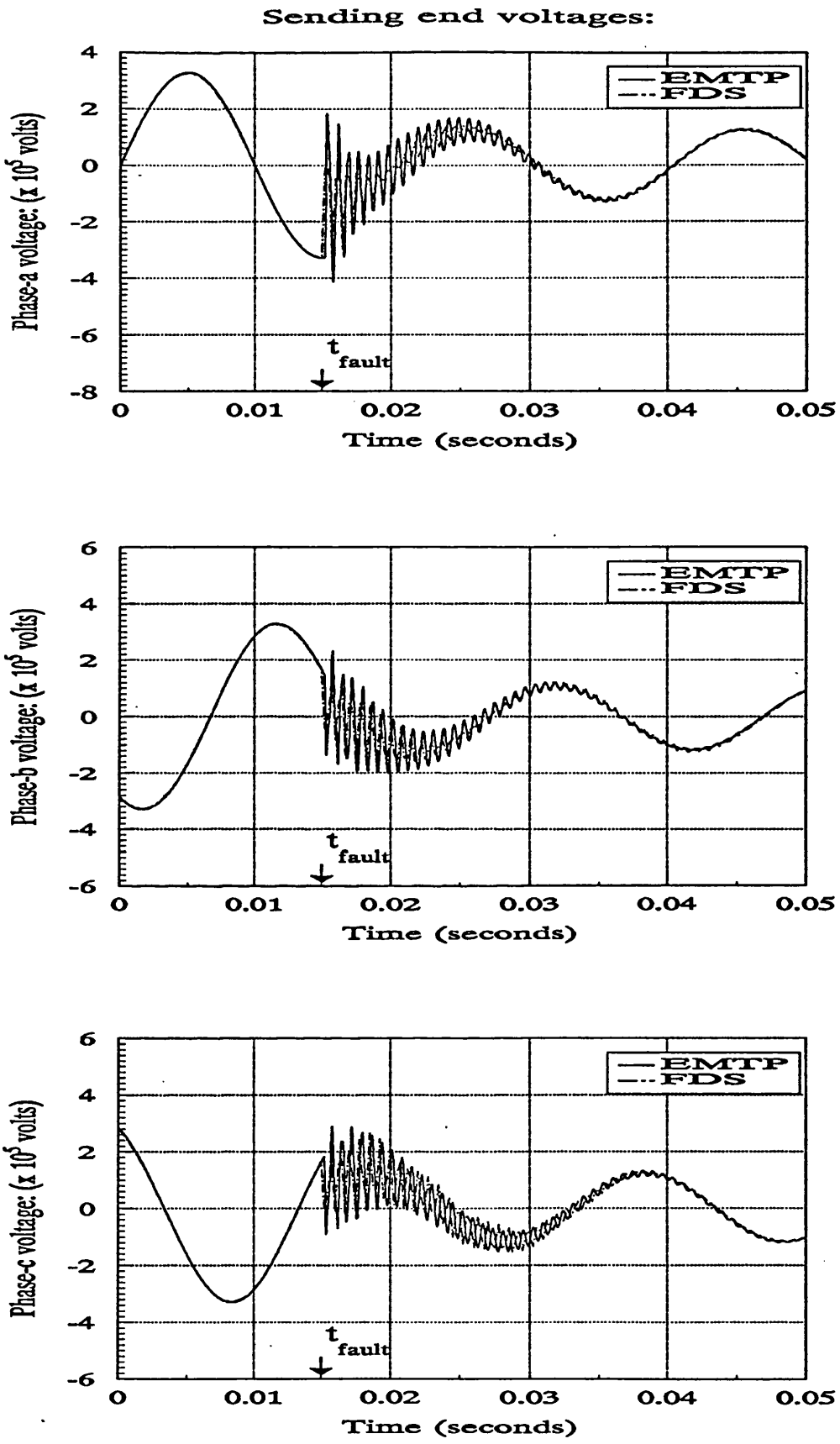

Figure 8.32: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

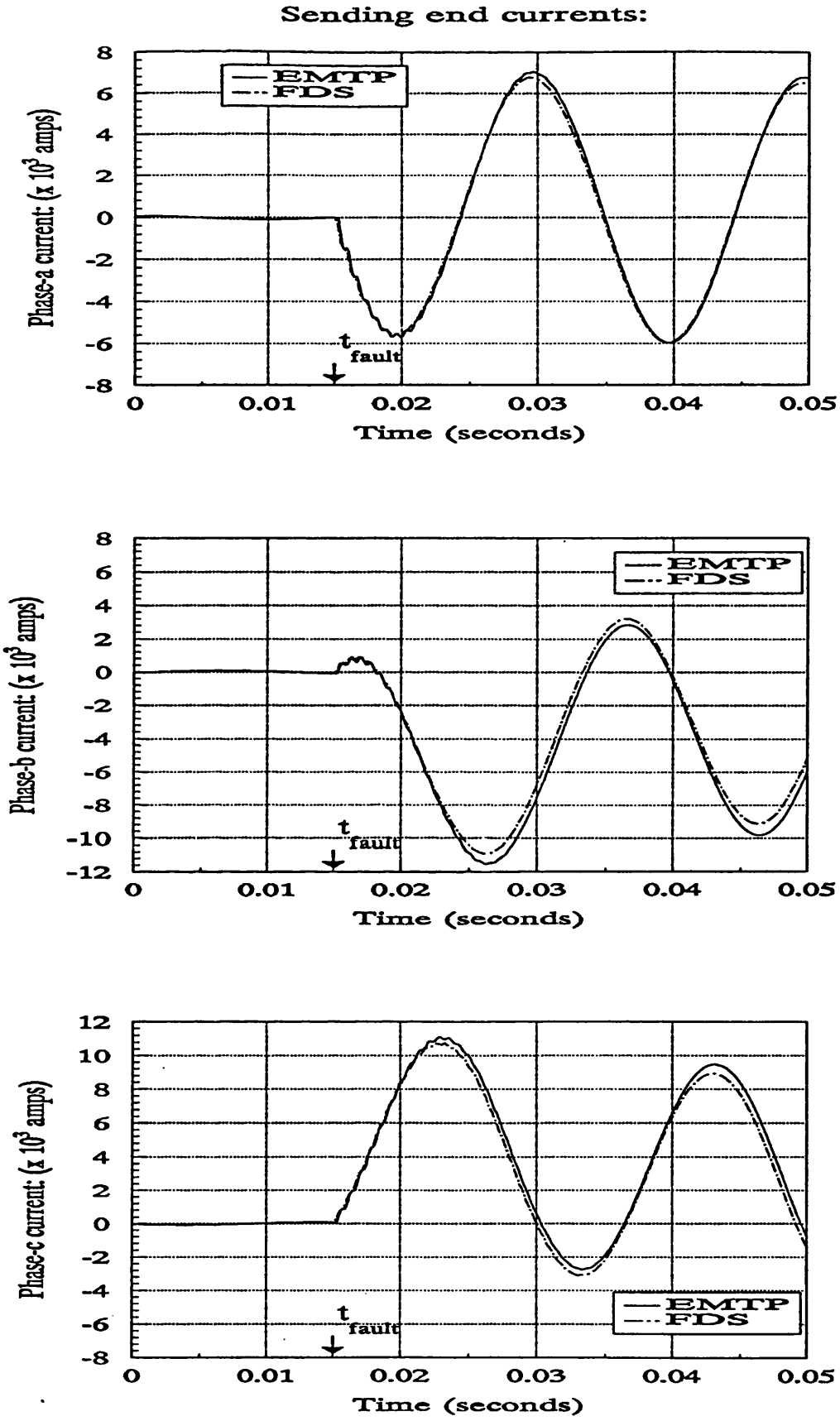

Figure 8.33: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

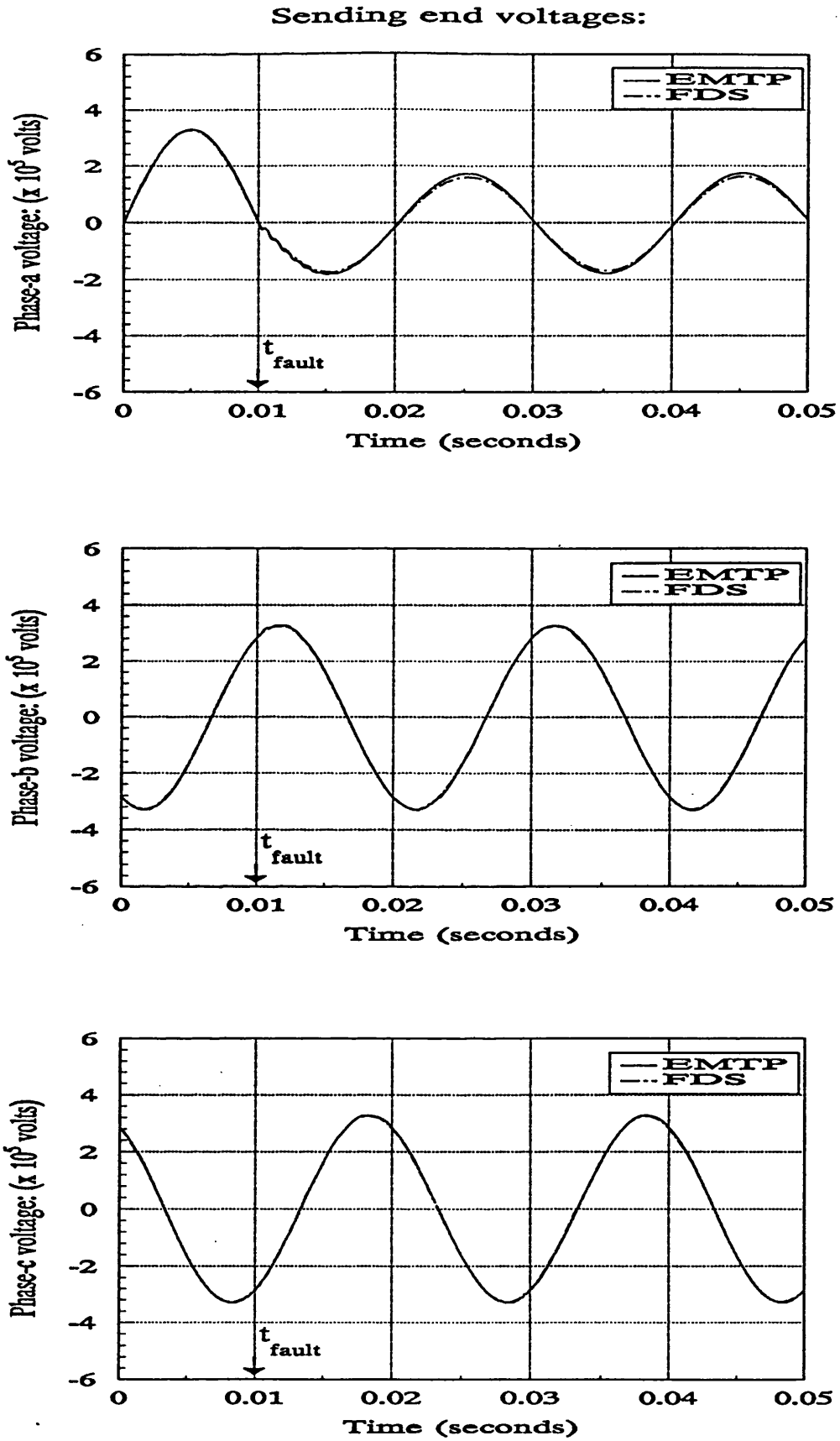

Figure 8.34: EMTP and FDS comparison: midpoint phase-a to earth fault, sending *s.c.l.* = 5GVA, receiving *s.c.l.* = 5GVA.

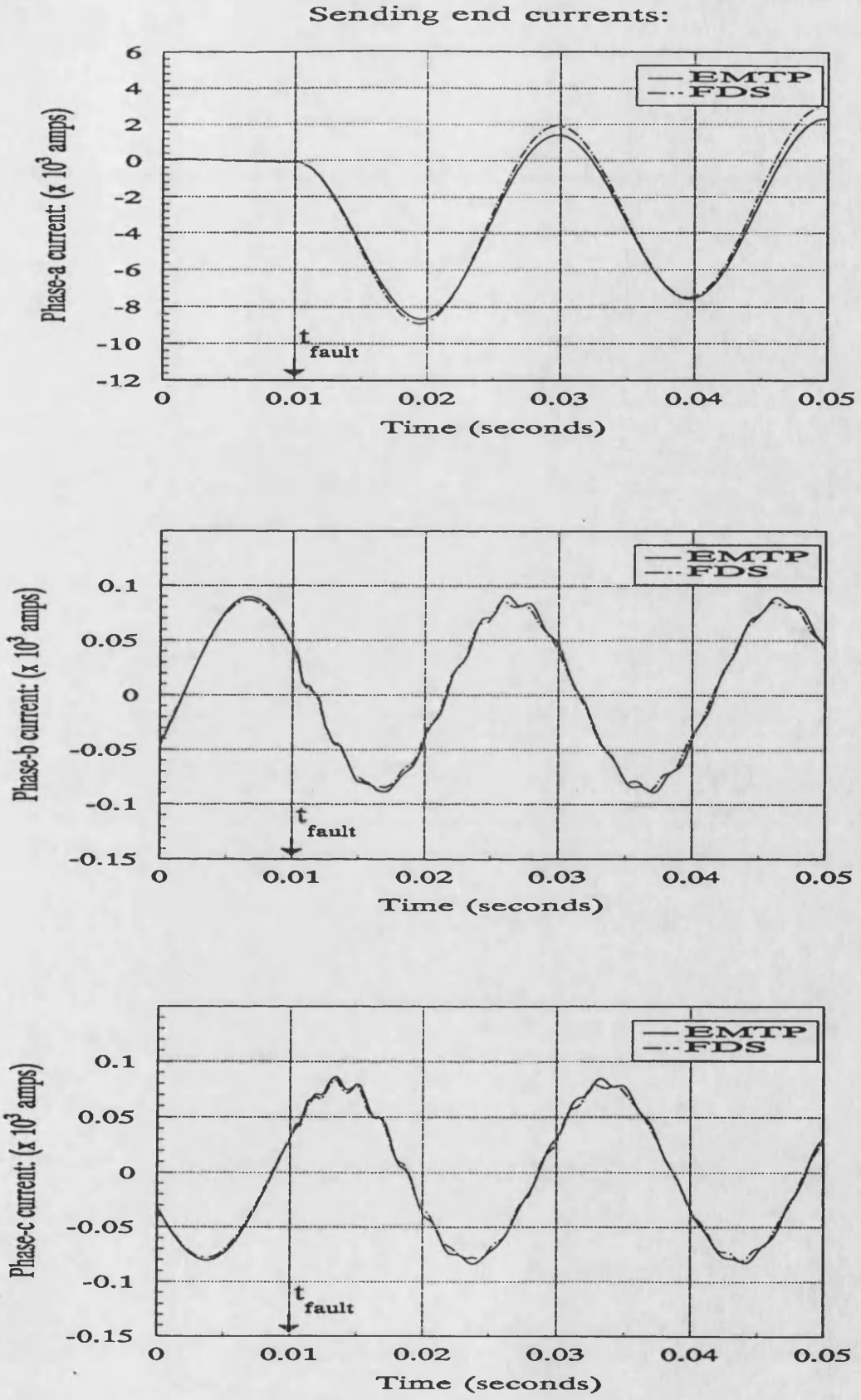

Figure 8.35: EMTP and FDS comparison: midpoint phase-a to earth fault, sending $s.c.l.$ = 5GVA, receiving $s.c.l.$ = 5GVA.

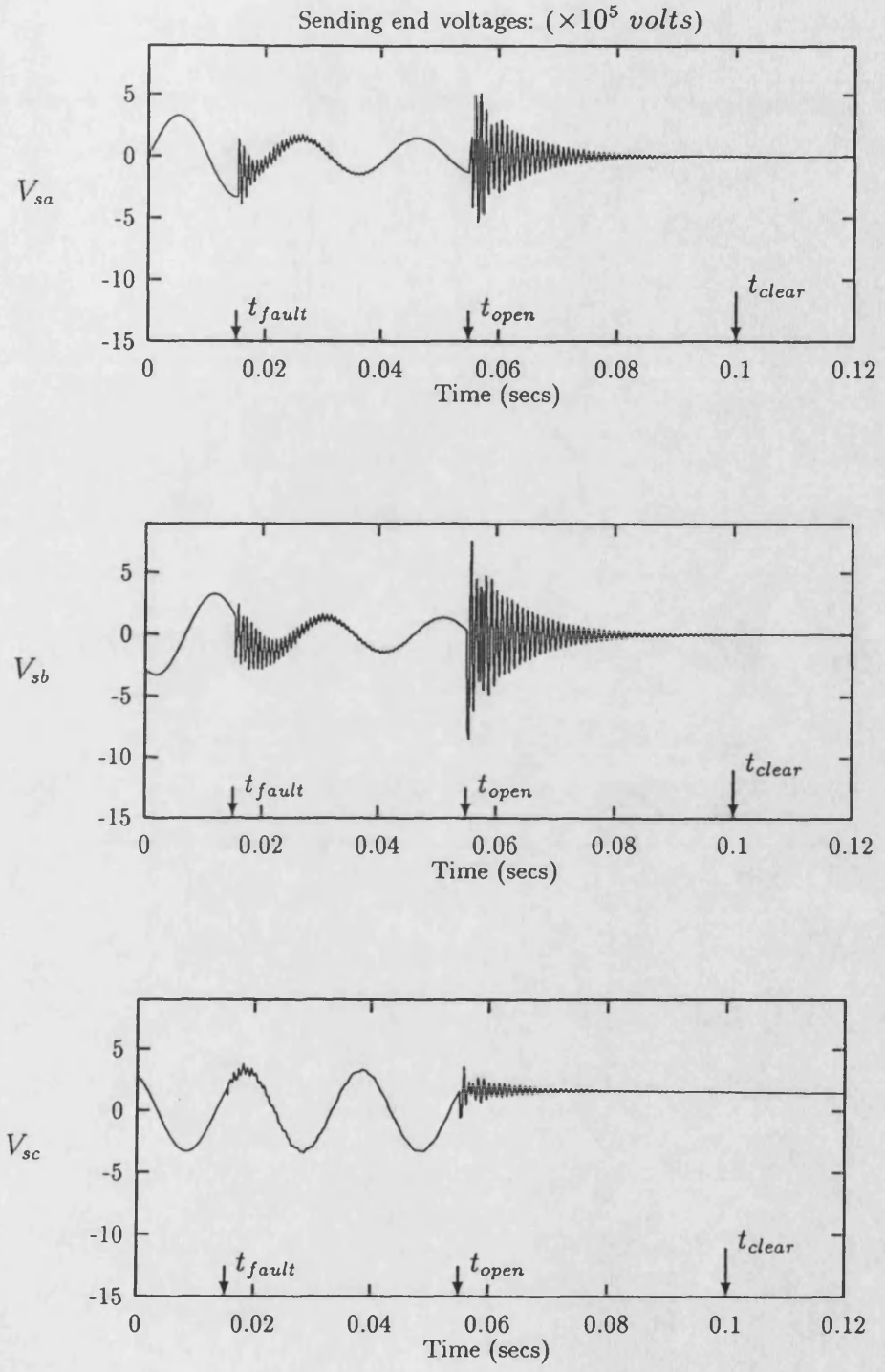

Figure 8.36: EMTP modelling: midpoint phase-ab to earth fault, breaker poles at both ends opened simultaneously.

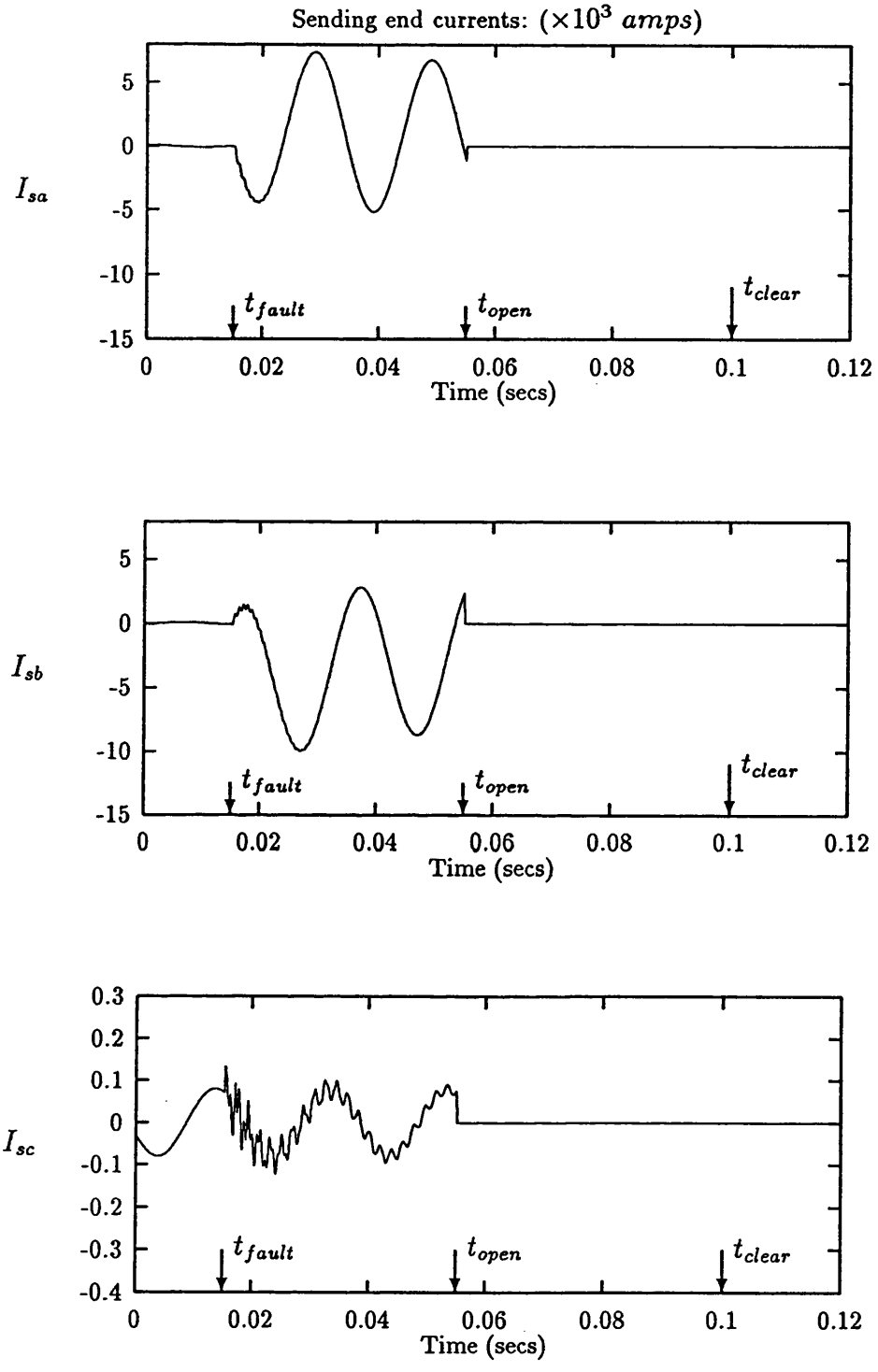

Figure 8.37: EMTP modelling: midpoint phase-ab to earth fault, breaker poles at both ends opened simultaneously.

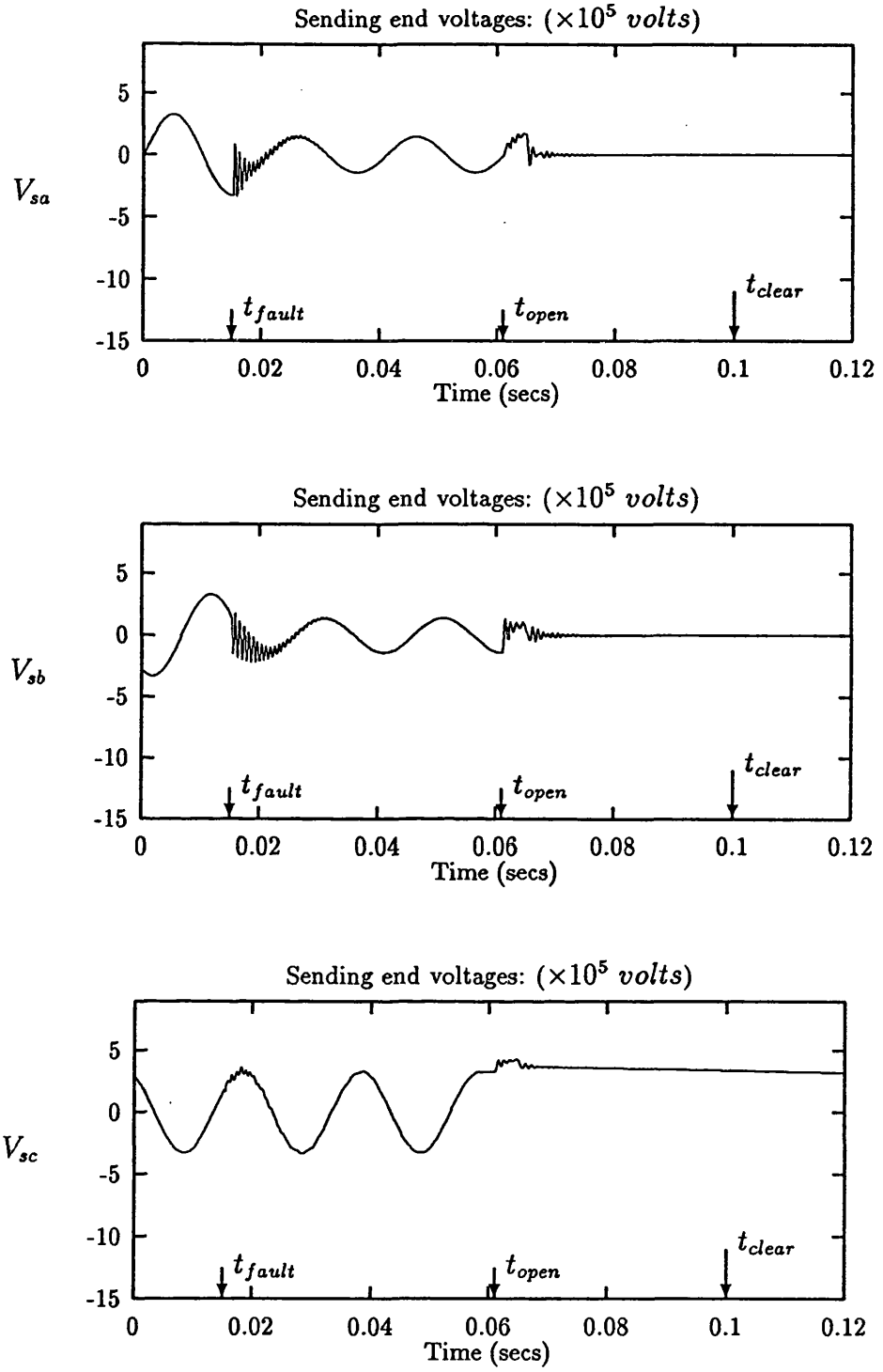

Figure 8.38: EMTP modelling: midpoint phase-ab to earth fault, breaker poles at both ends opened sequentially.

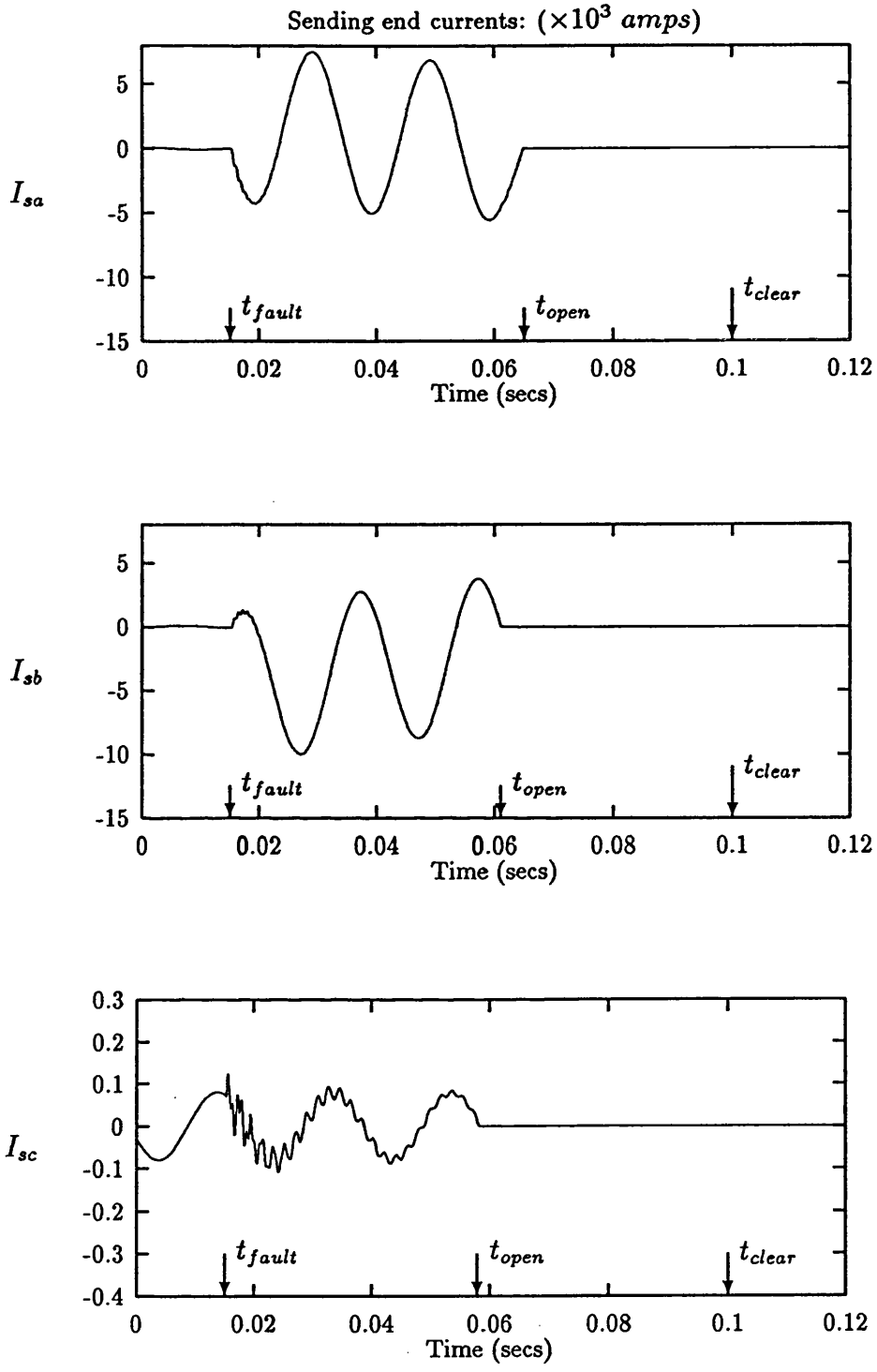

Figure 8.39: EMTP modelling: midpoint phase-ab to earth fault, breaker poles at both ends opened sequentially.

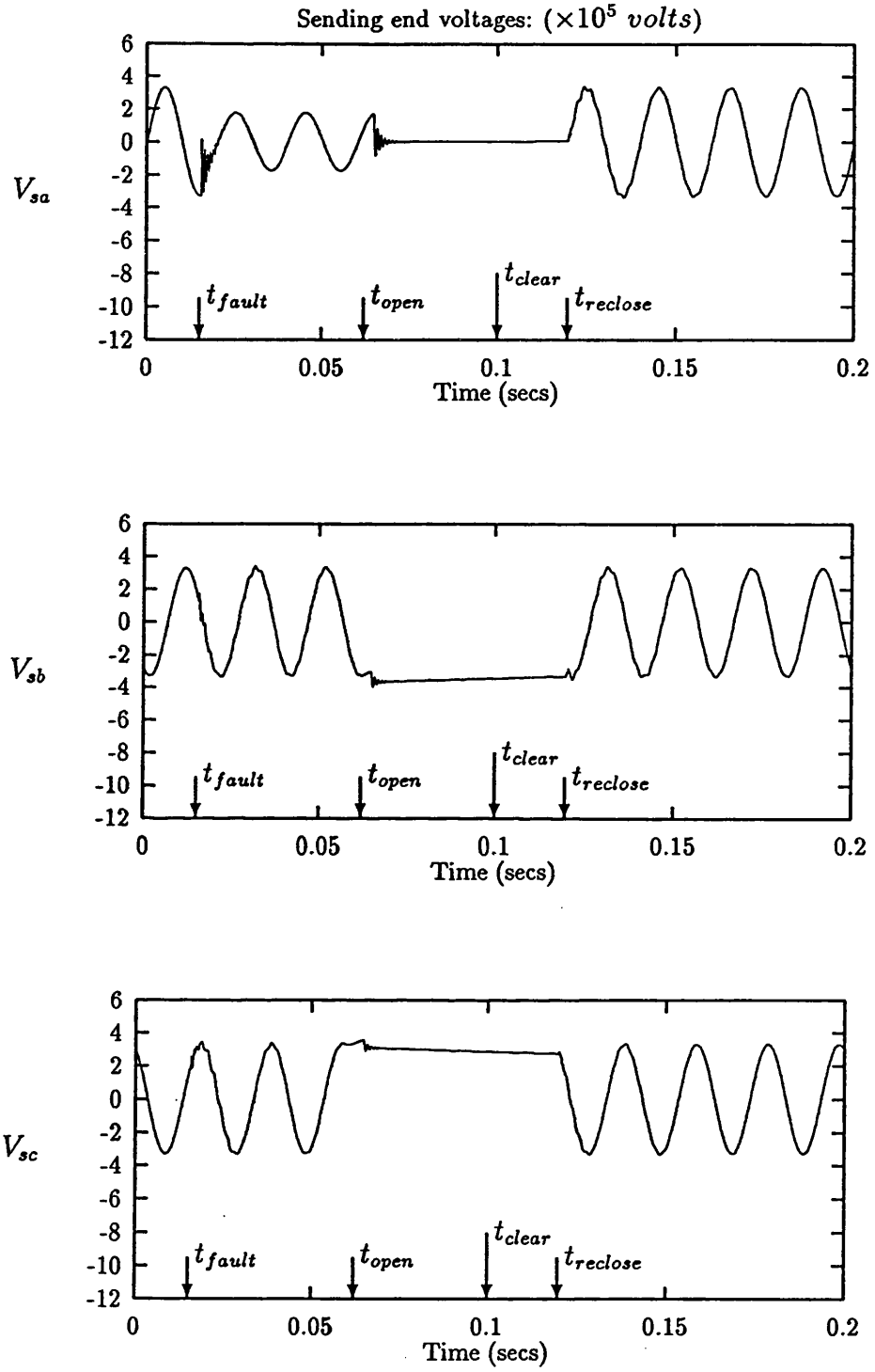

Figure 8.40: EMTP modelling: three phase autoreclosure sequence for midpoint phase-a to earth fault, breaker poles at both ends opened sequentially.

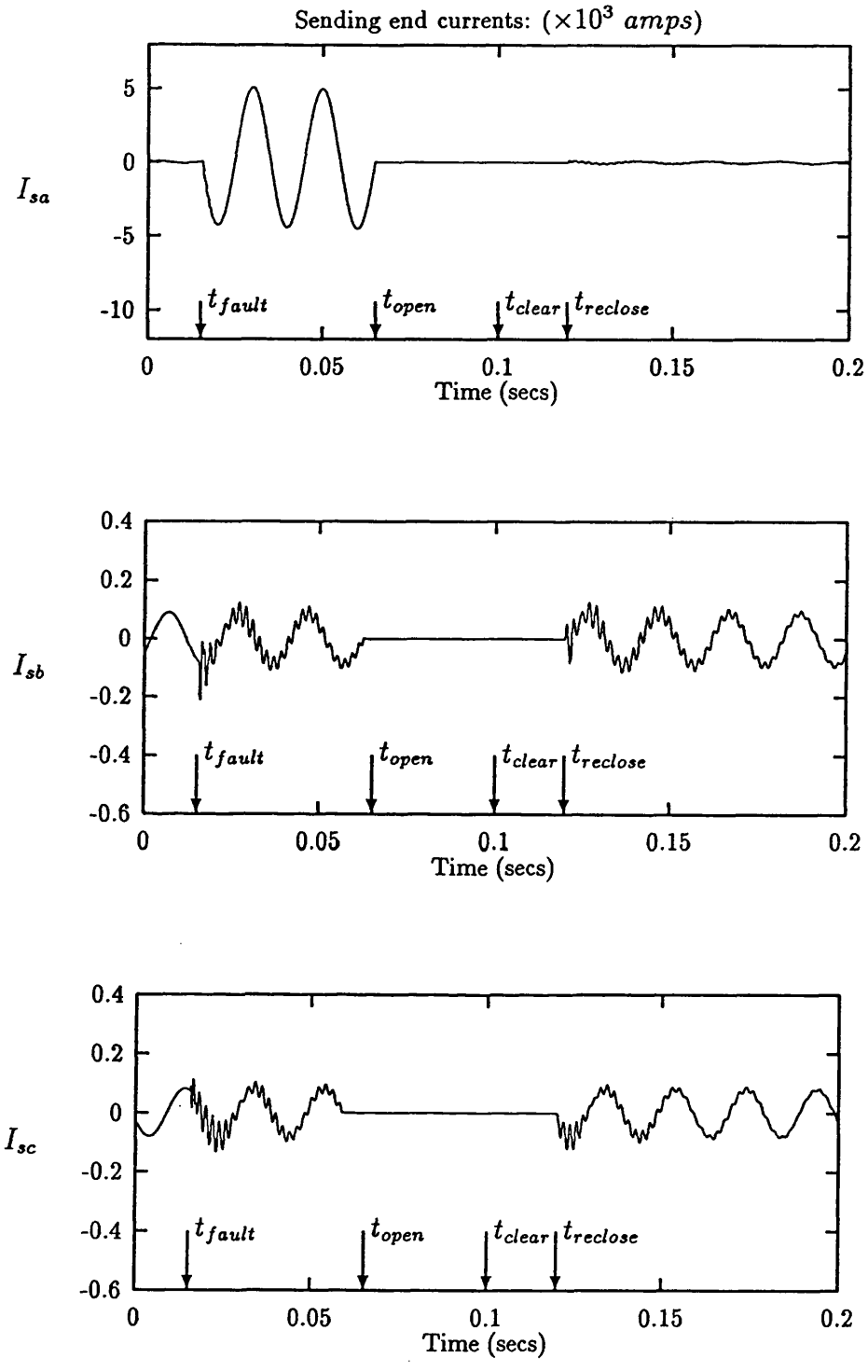

Figure 8.41: EMTP modelling: three phase autoreclosure sequence for midpoint phase-a to earth fault, breaker poles at both ends opened sequentially.

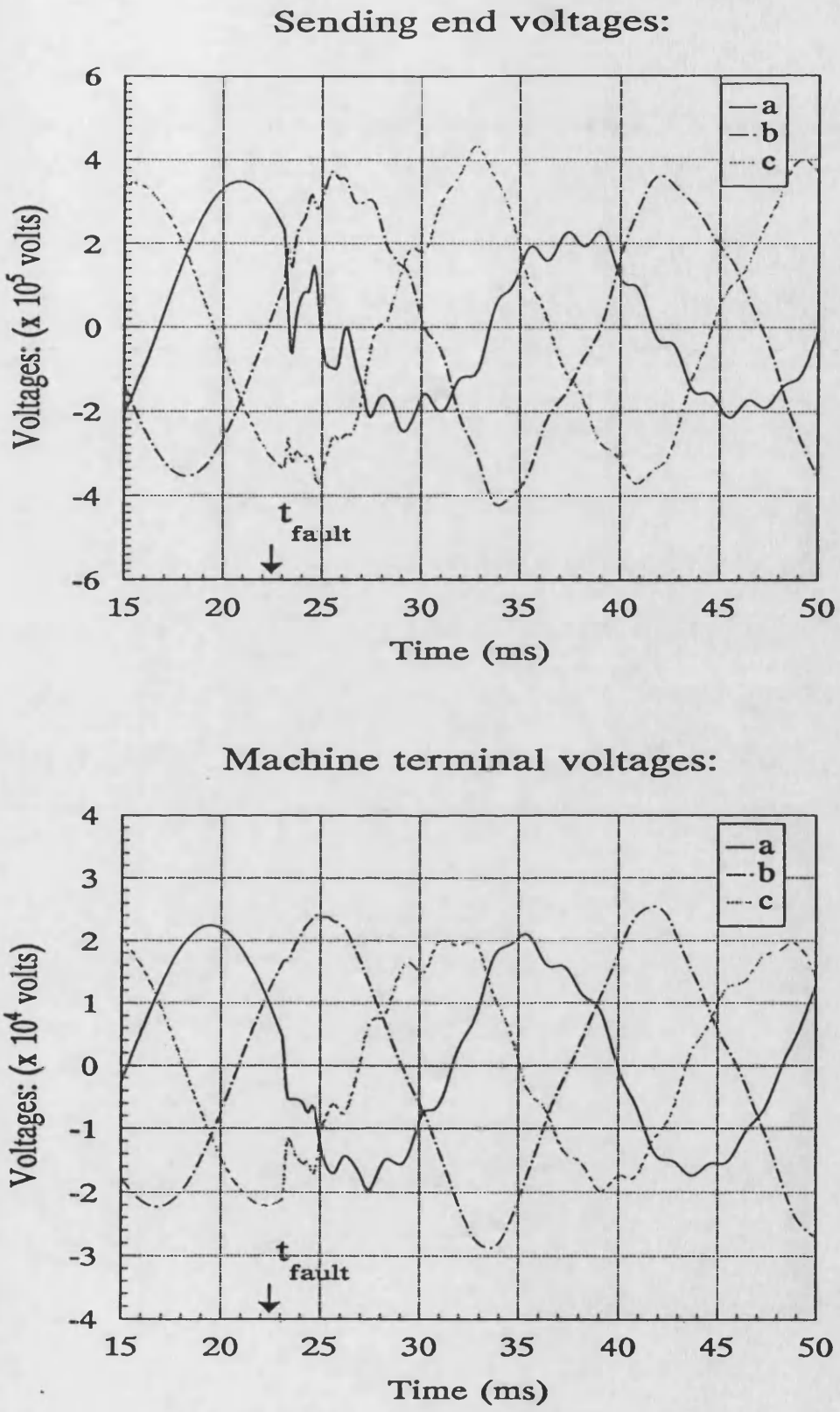

Figure 8.42: Actual source model: midpoint a-earth fault, sending end voltages (top), machine terminal voltages (bottom).

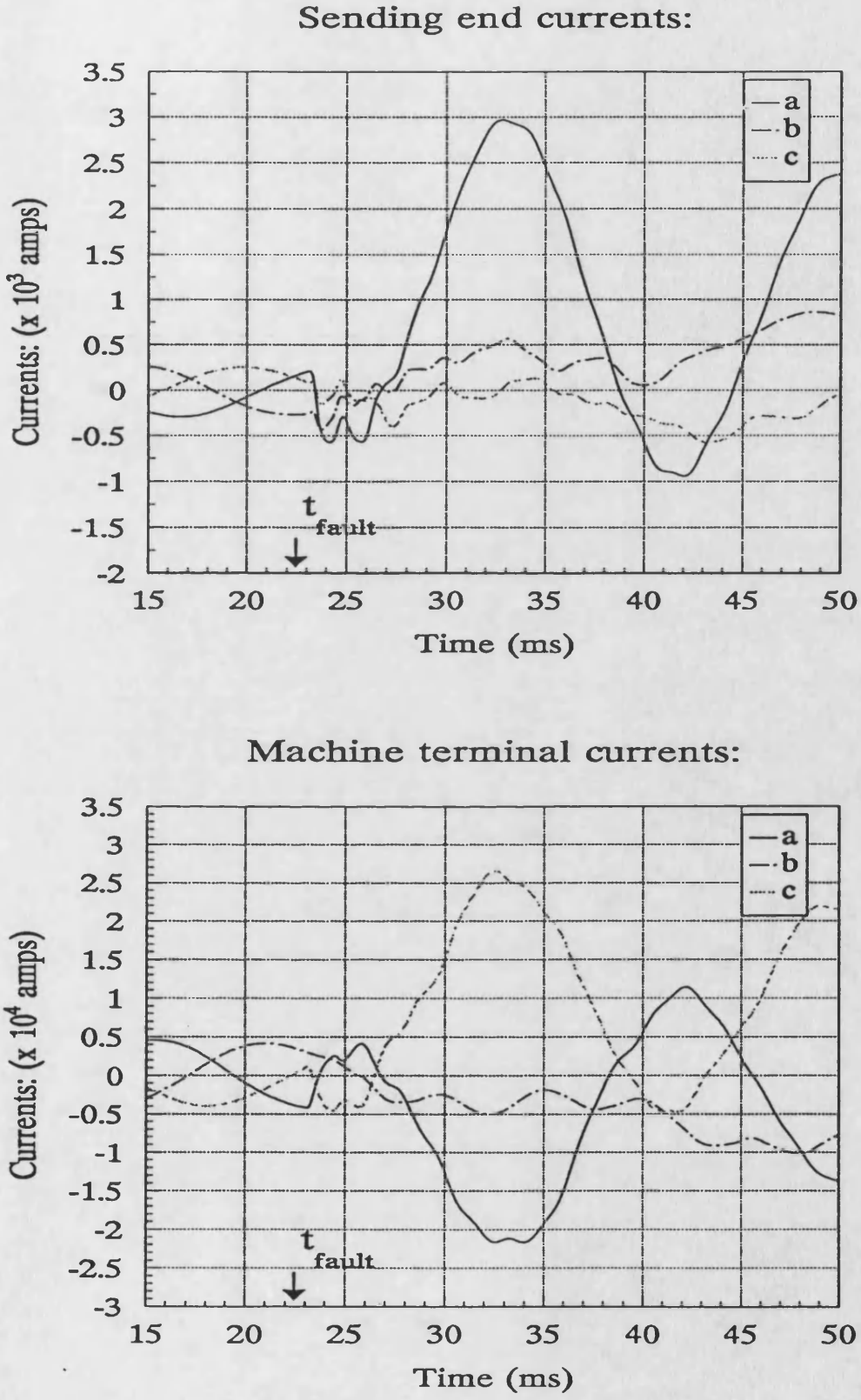

Figure 8.43: Actual source model: midpoint a-earth fault, sending end currents (top), currents at machine terminals (bottom).

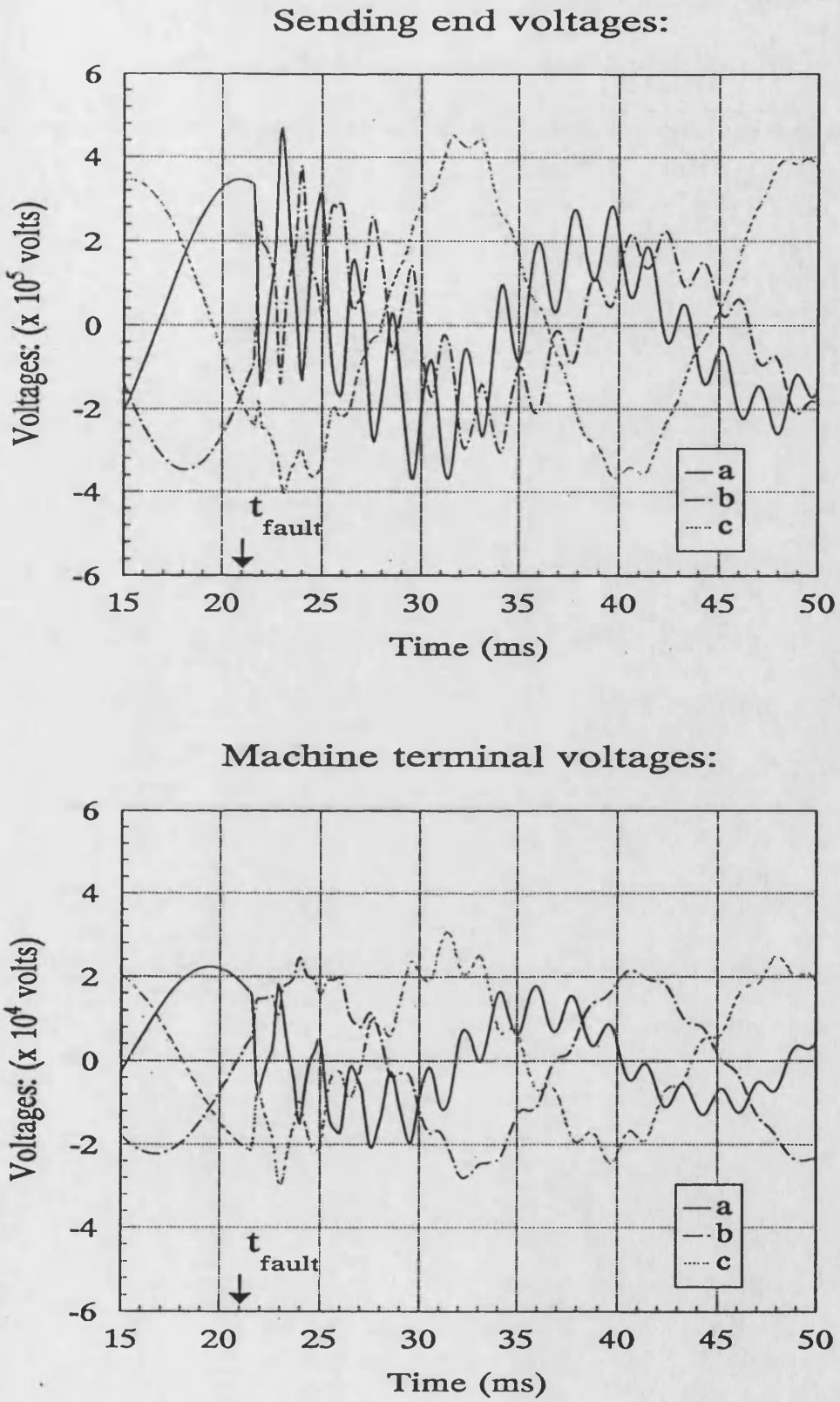

Figure 8.44: Actual source model: midpoint ab-earth fault, sending end voltages (top), machine terminal voltages (bottom).

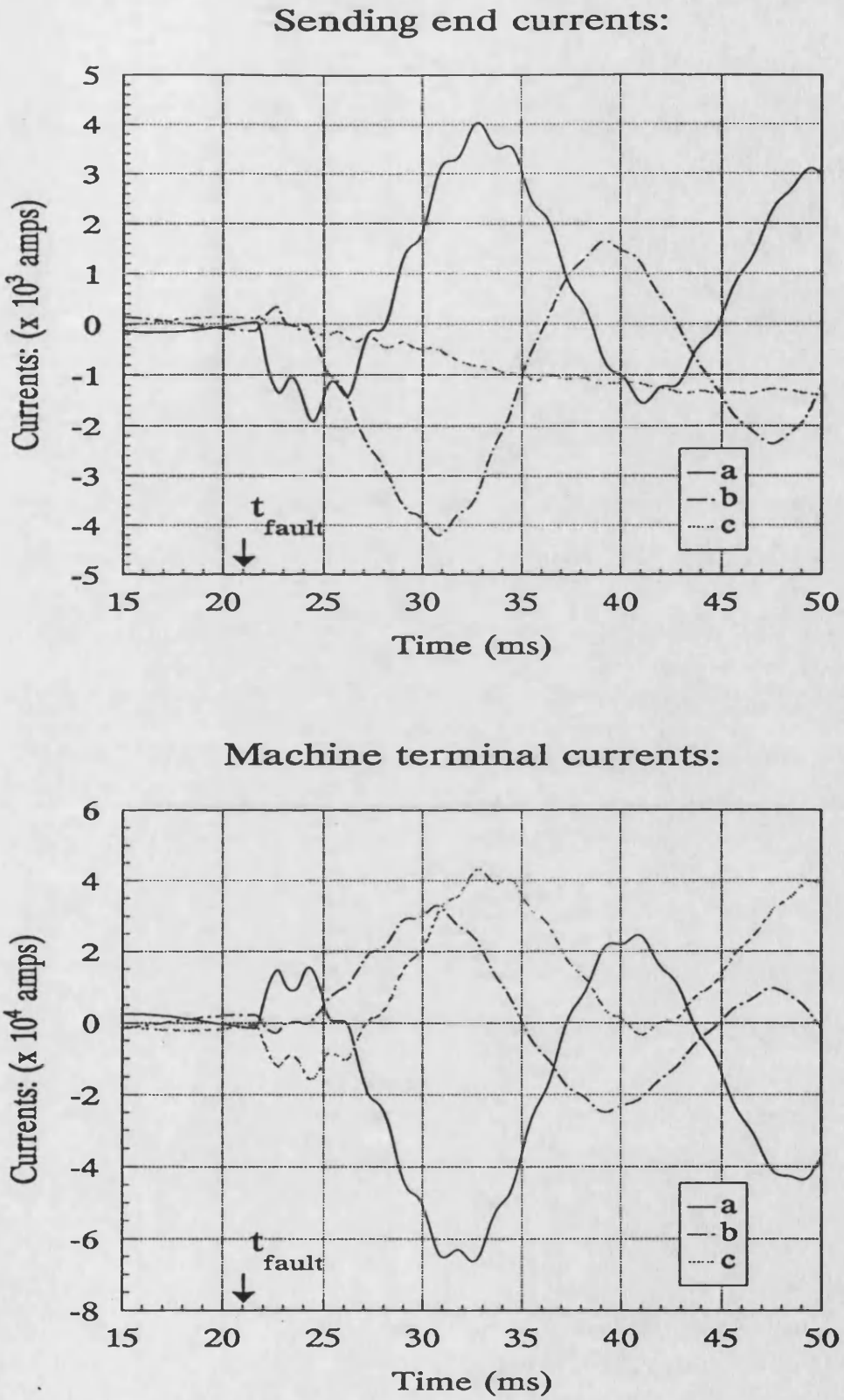

Figure 8.45: Actual source model: midpoint ab-earth fault, sending end currents (top), currents at machine terminals (bottom).

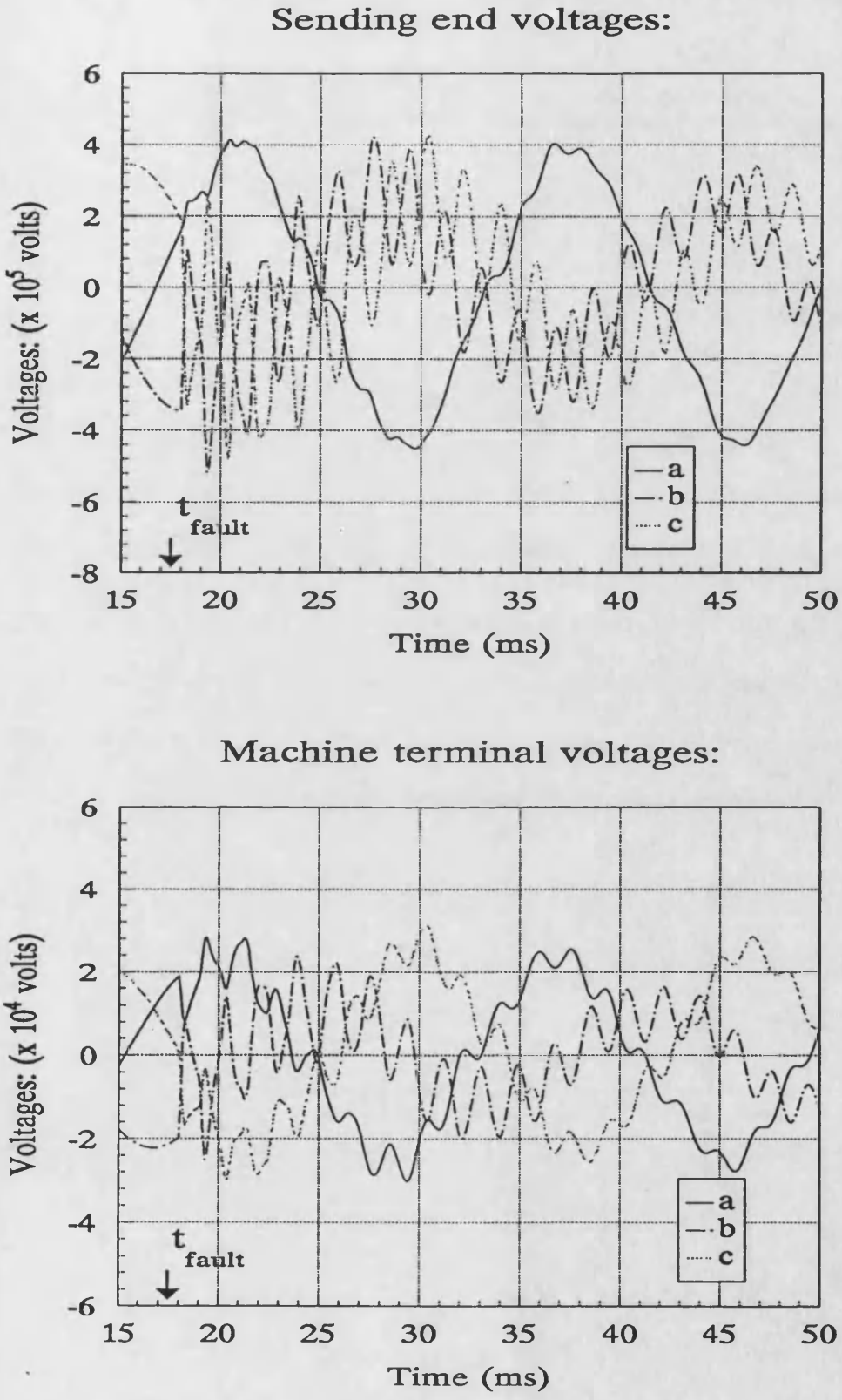

Figure 8.46: Actual source model: midpoint bc-phase fault, sending end voltages (top), machine terminal voltages (bottom).

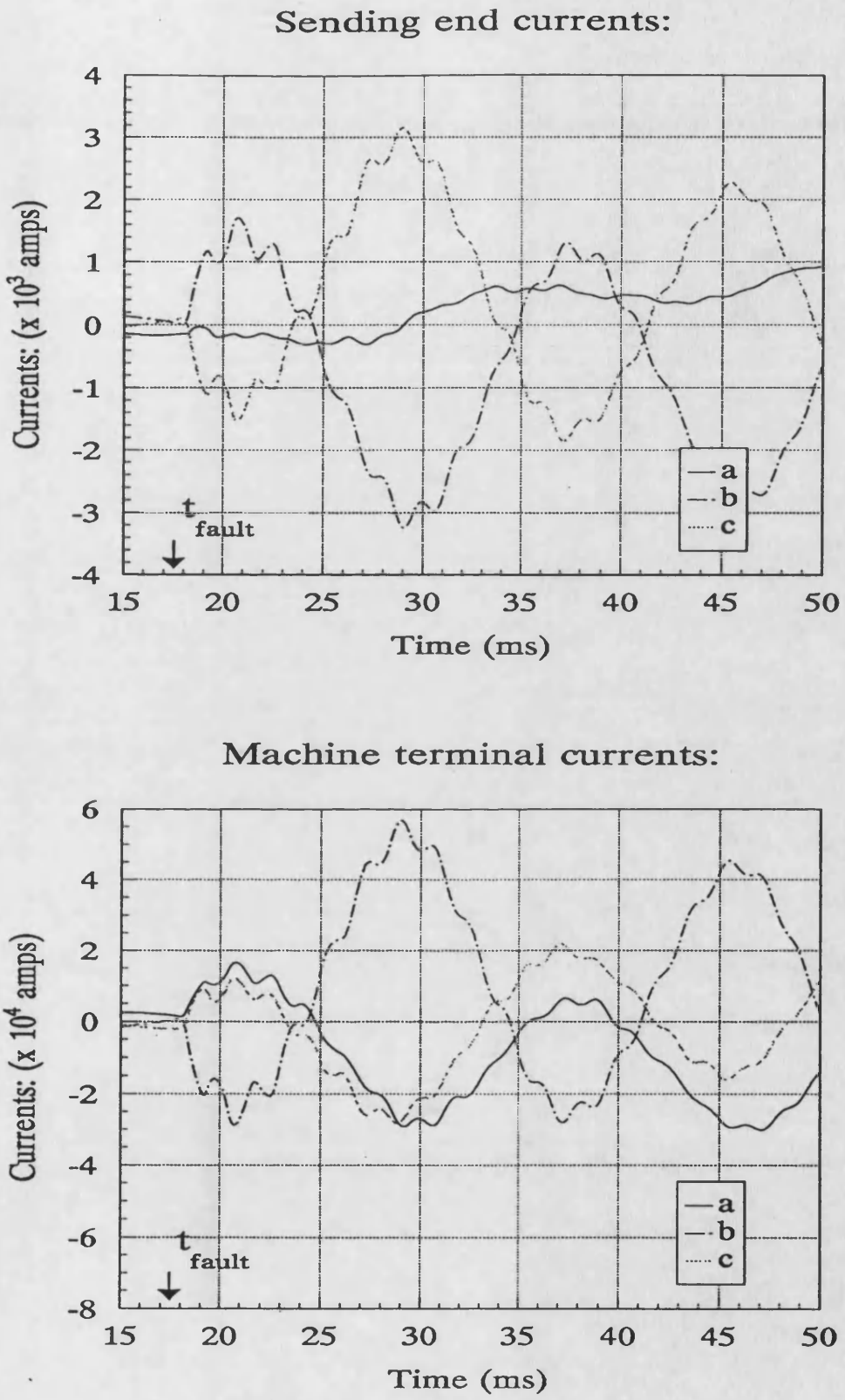

Figure 8.47: Actual source model: midpoint bc-phase fault, sending end currents (top), currents at machine terminals (bottom).

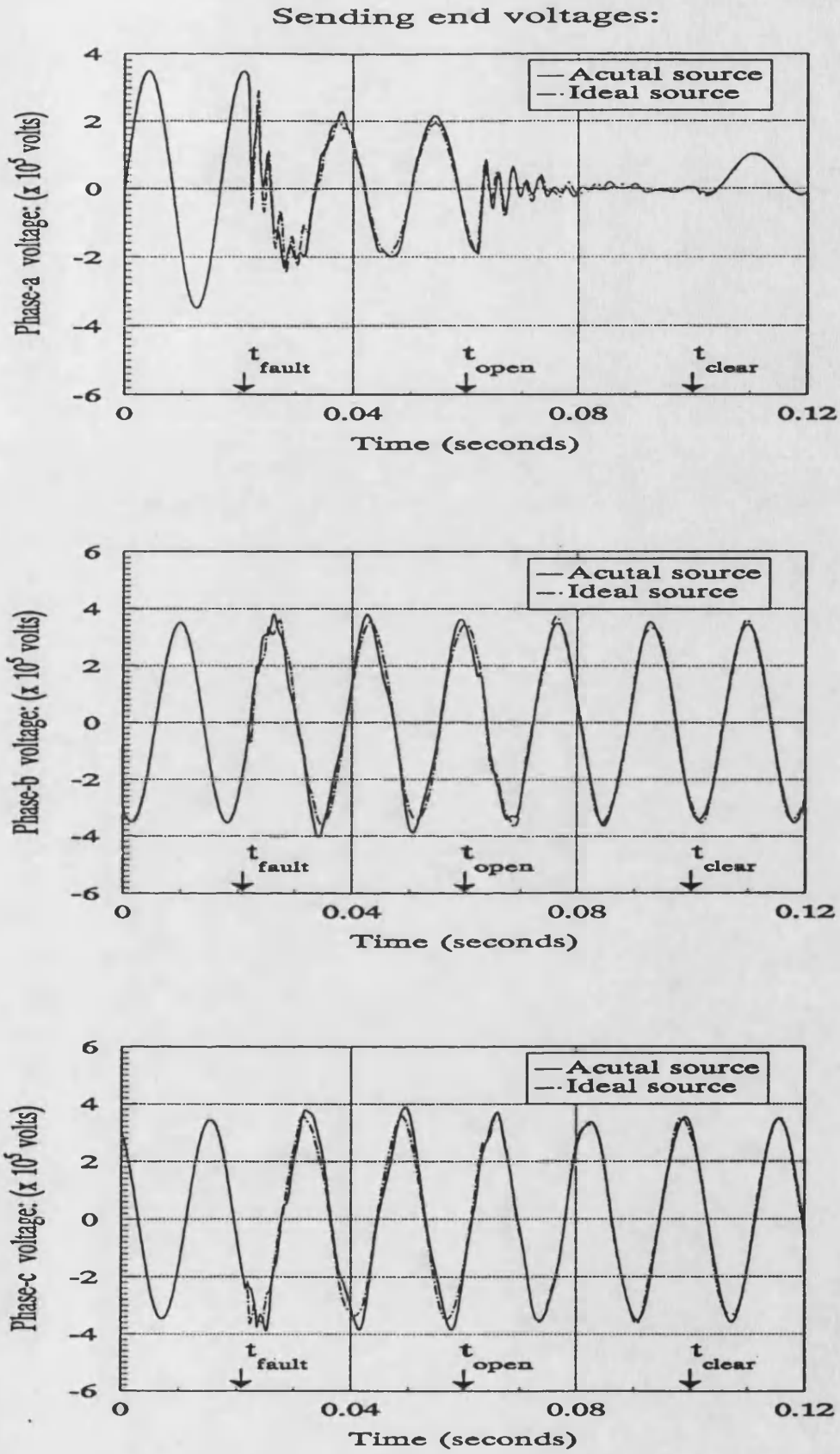

Figure 8.48: Actual and ideal source model comparison: midpoint a-earth near voltage maximum fault.

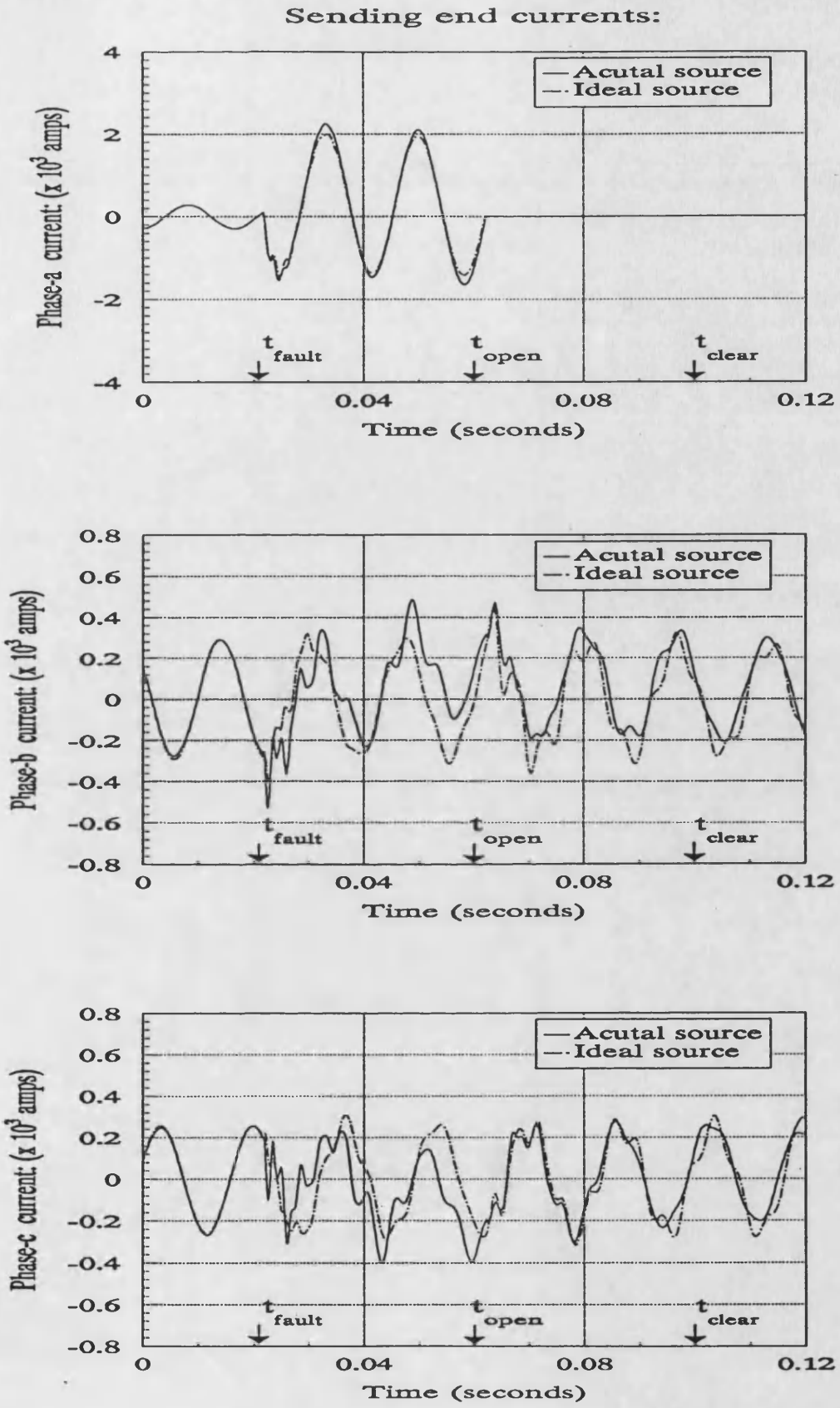

Figure 8.49: Actual and ideal source model comparison: midpoint a-earth near voltage maximum fault.

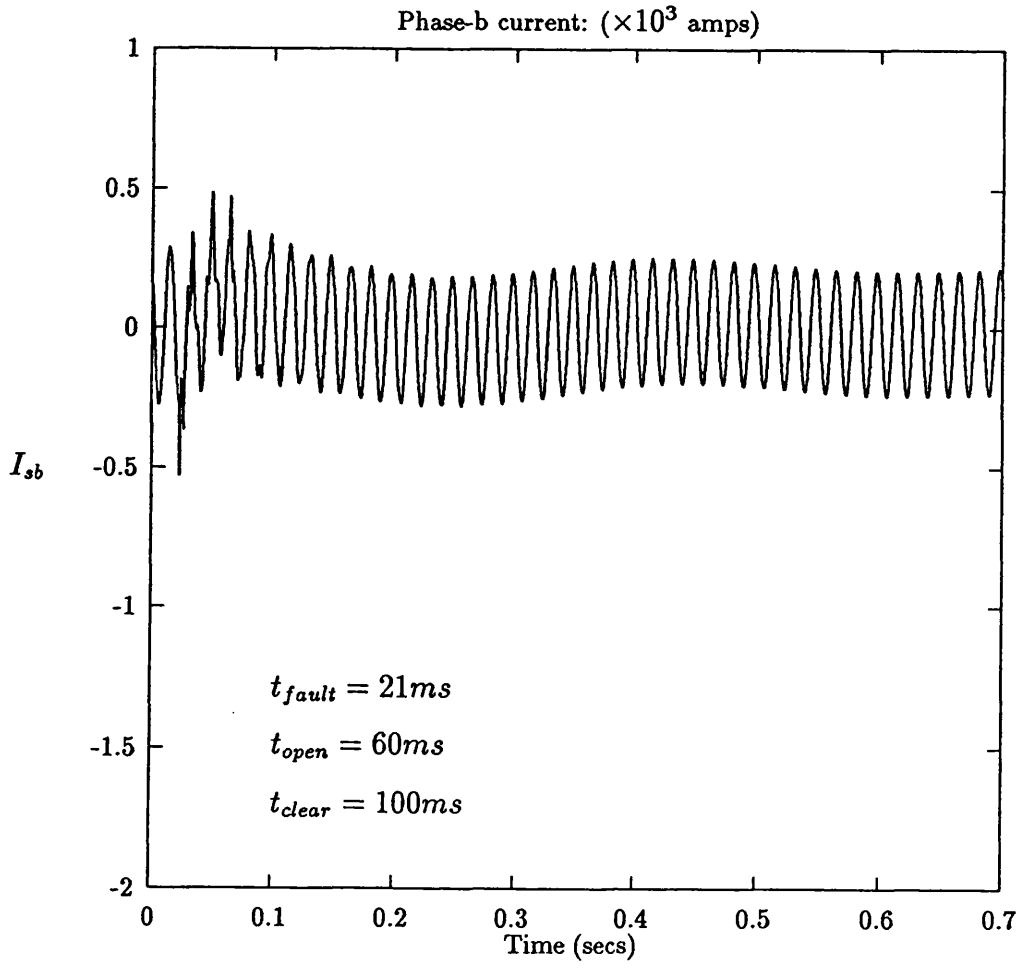

Figure 8.50: Actual source model: midpoint a-earth near voltage maximum fault, extended simulation for sending end b-phase current.

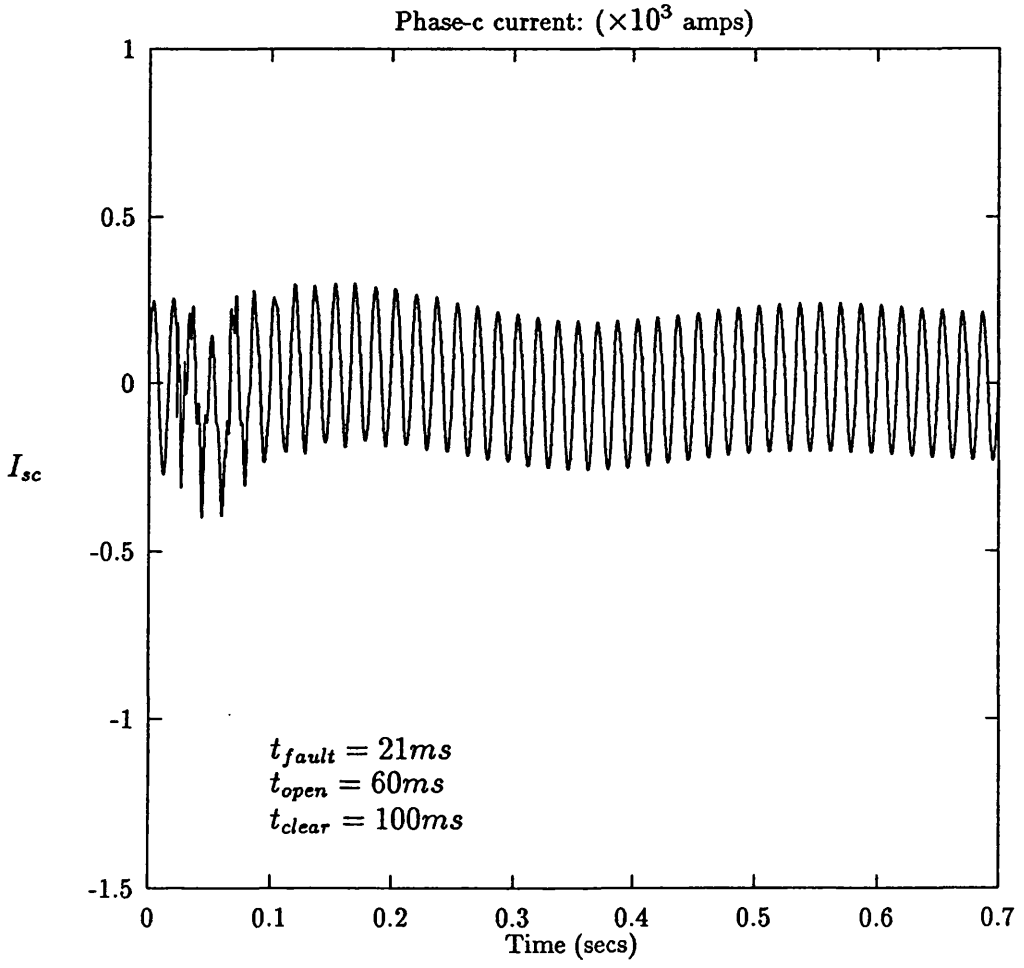

Figure 8.51: Actual source model: midpoint a-earth near voltage maximum fault, extended simulation for sending end c-phase current.

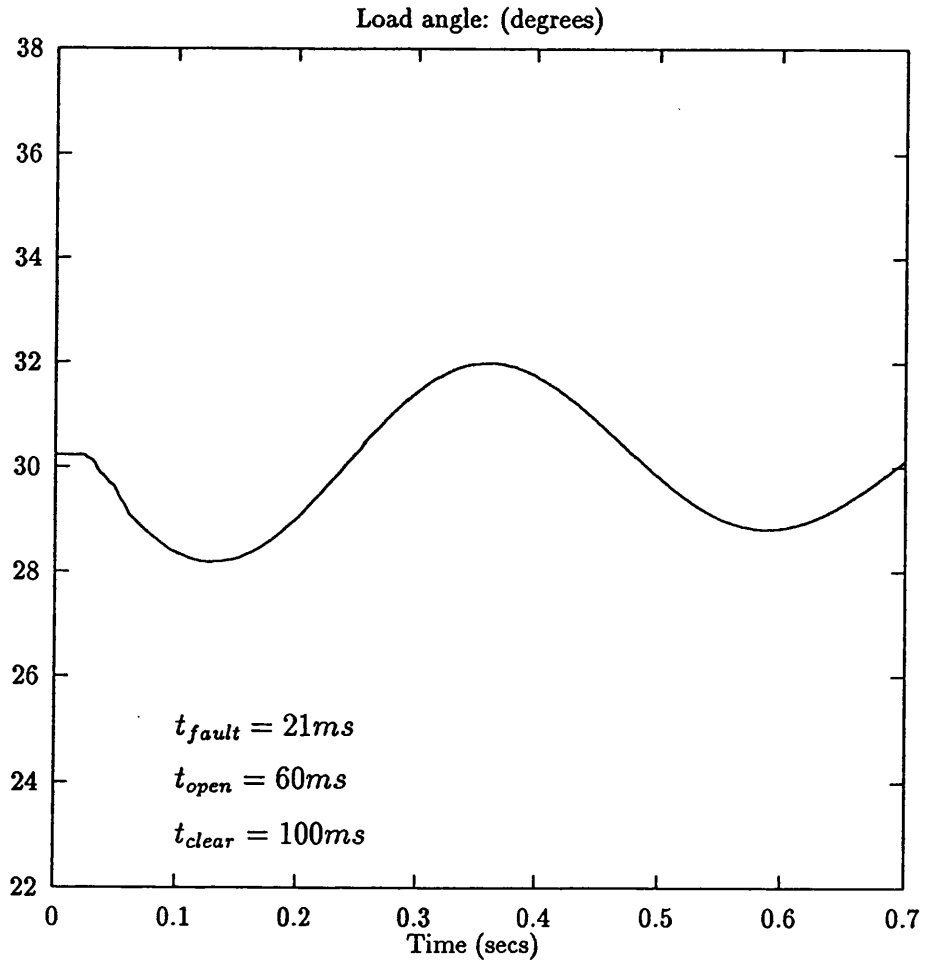

Figure 8.52: Actual source model: midpoint a-earth near voltage maximum fault, load angle change.

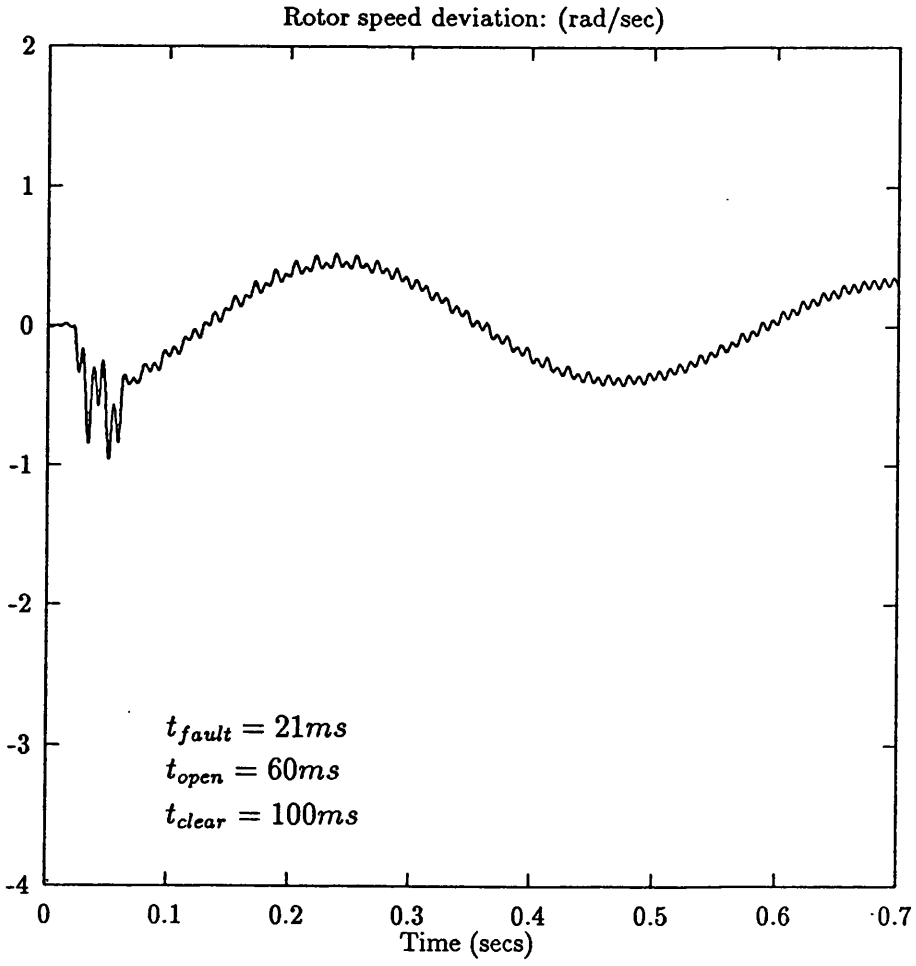

Figure 8.53: Actual source model: midpoint a-earth near voltage maximum fault, rotor speed deviations.

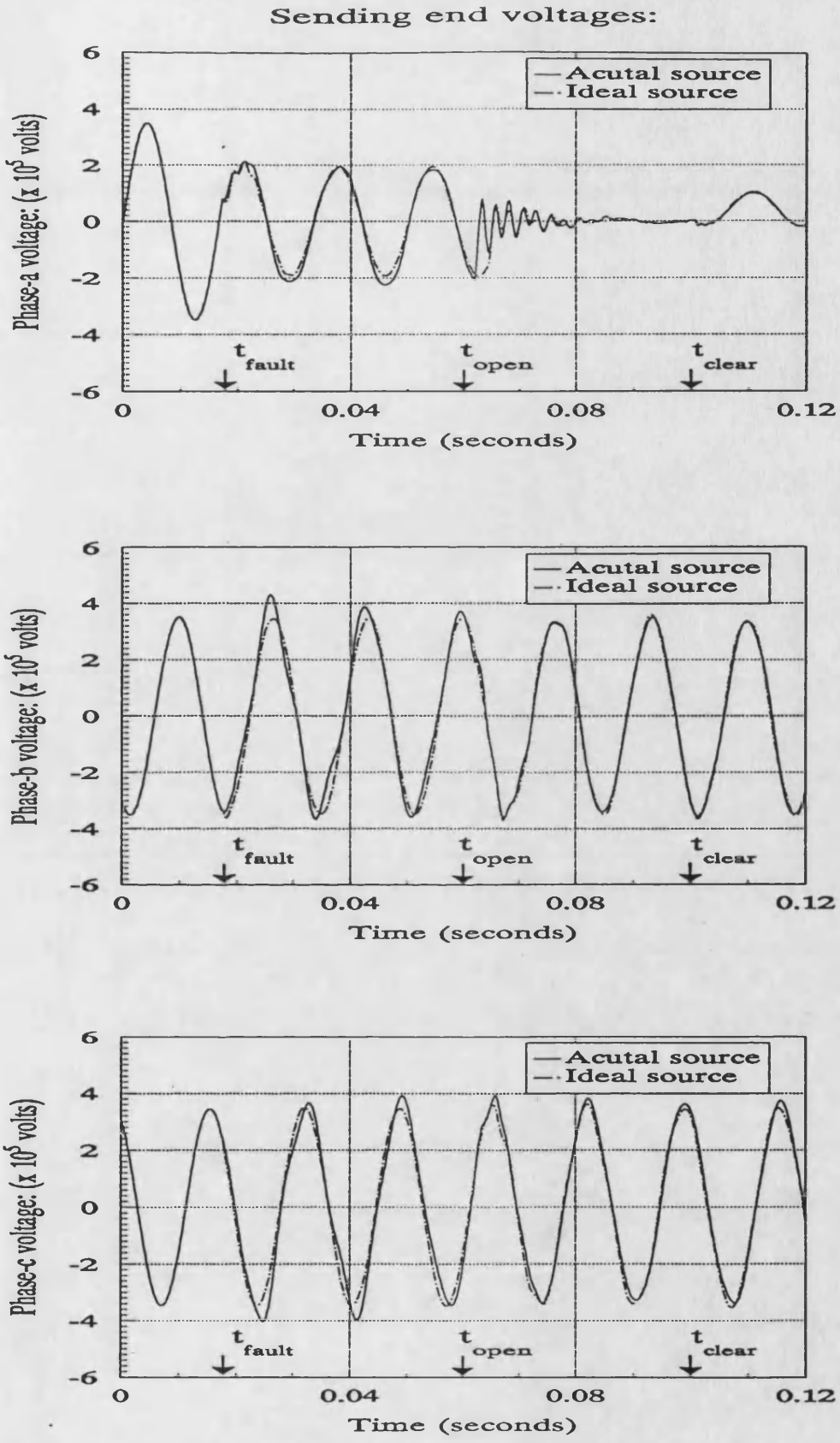

Figure 8.54: Actual and ideal source model comparison: midpoint a-earth near voltage zero fault.

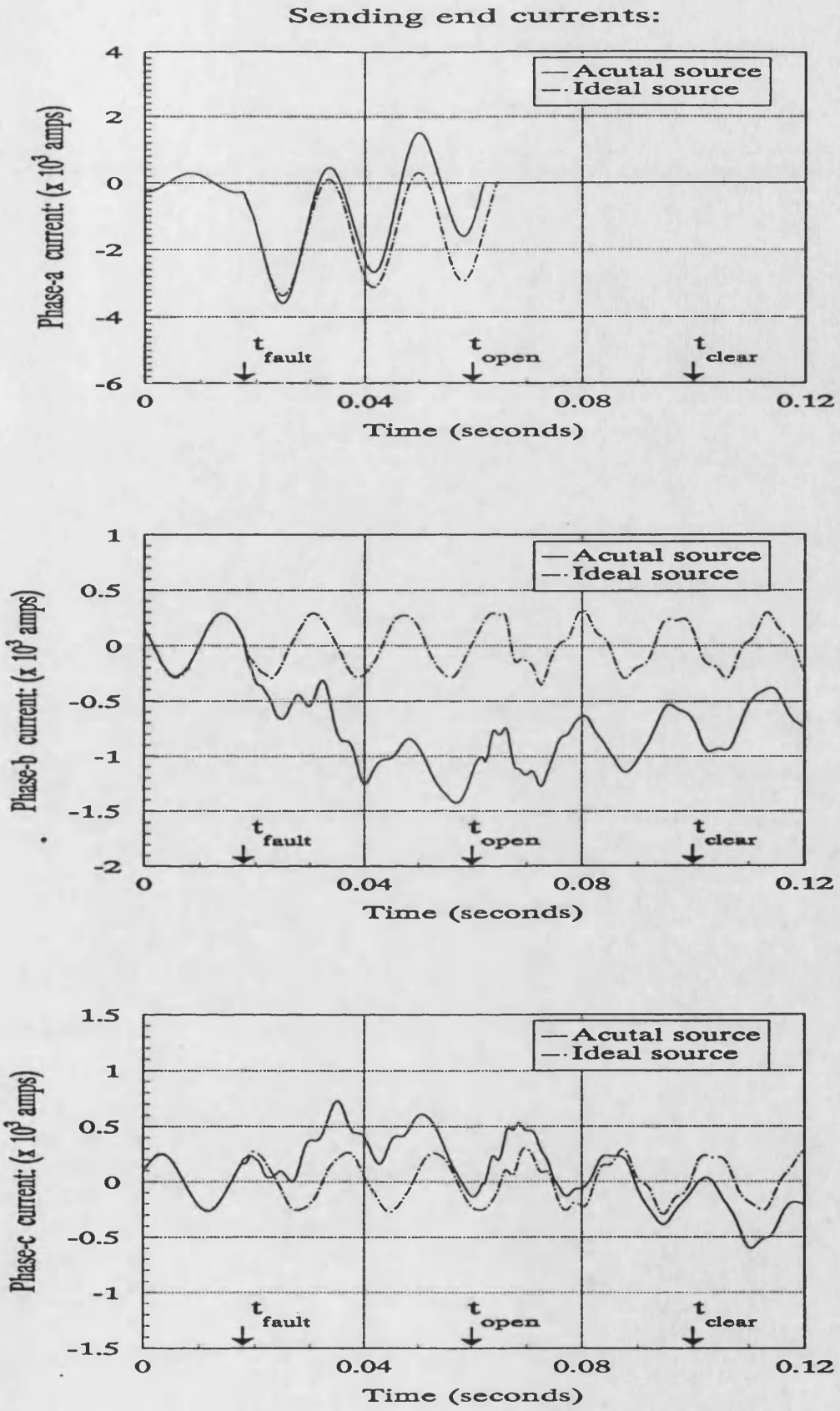

Figure 8.55: Actual and ideal source model comparison: midpoint a-earth near voltage zero fault.

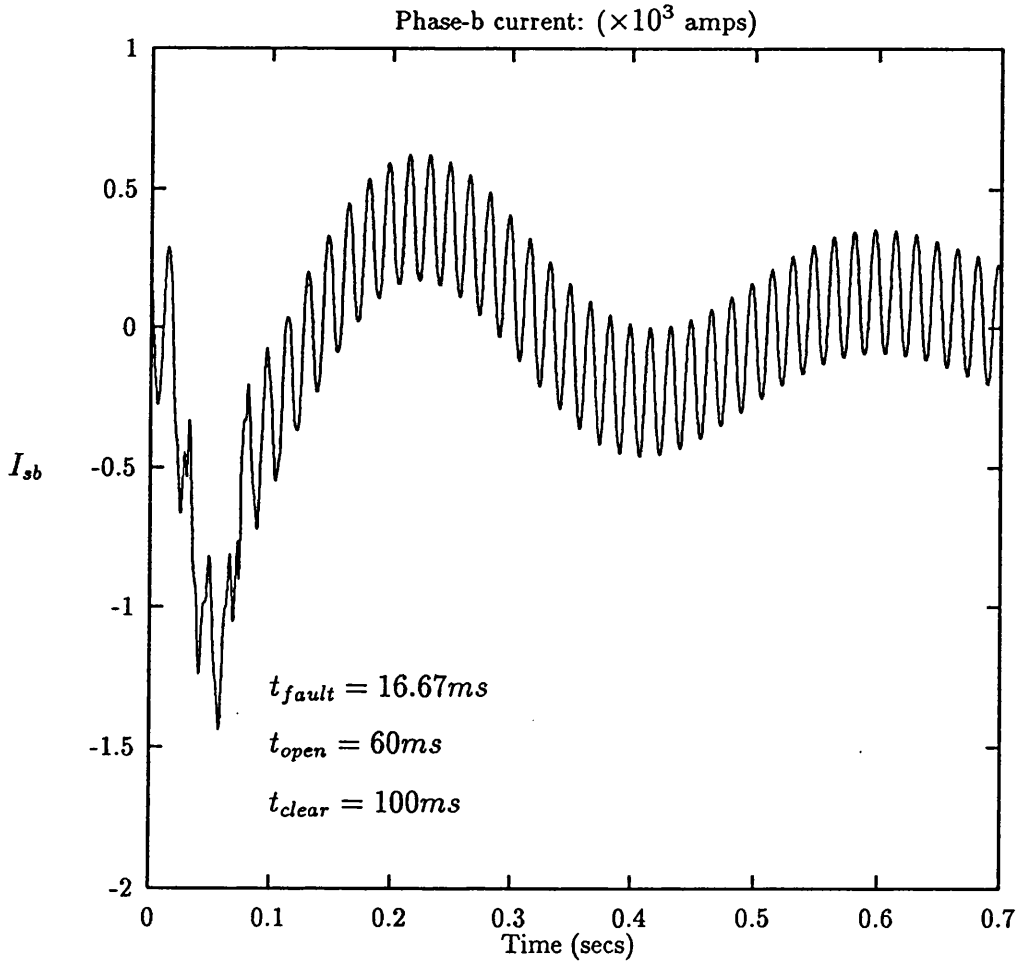

Figure 8.56: Actual source model: midpoint a-earth near voltage zero fault, extended simulation for sending end b-phase current.

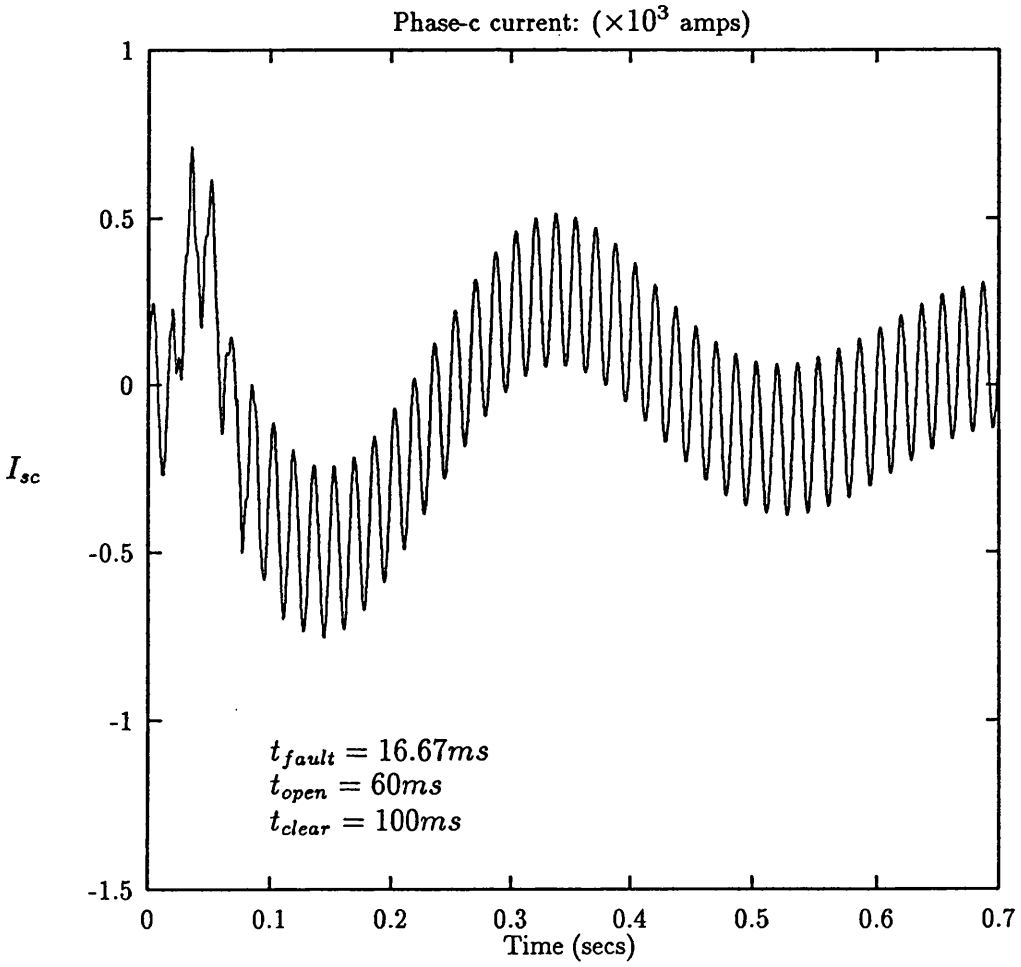

Figure 8.57: Actual source model: midpoint a-earth near voltage zero fault, extended simulation for sending end c-phase current.

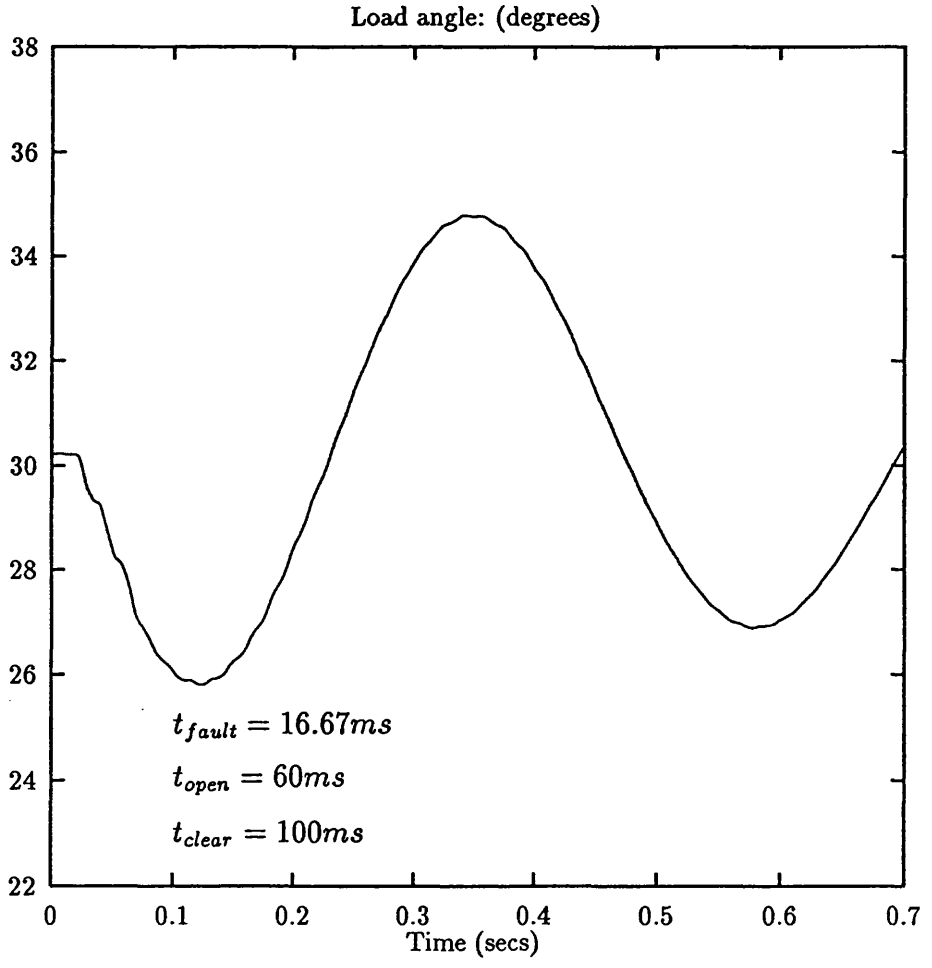

Figure 8.58: Actual source model: midpoint a-earth near voltage zero fault, load angle change.

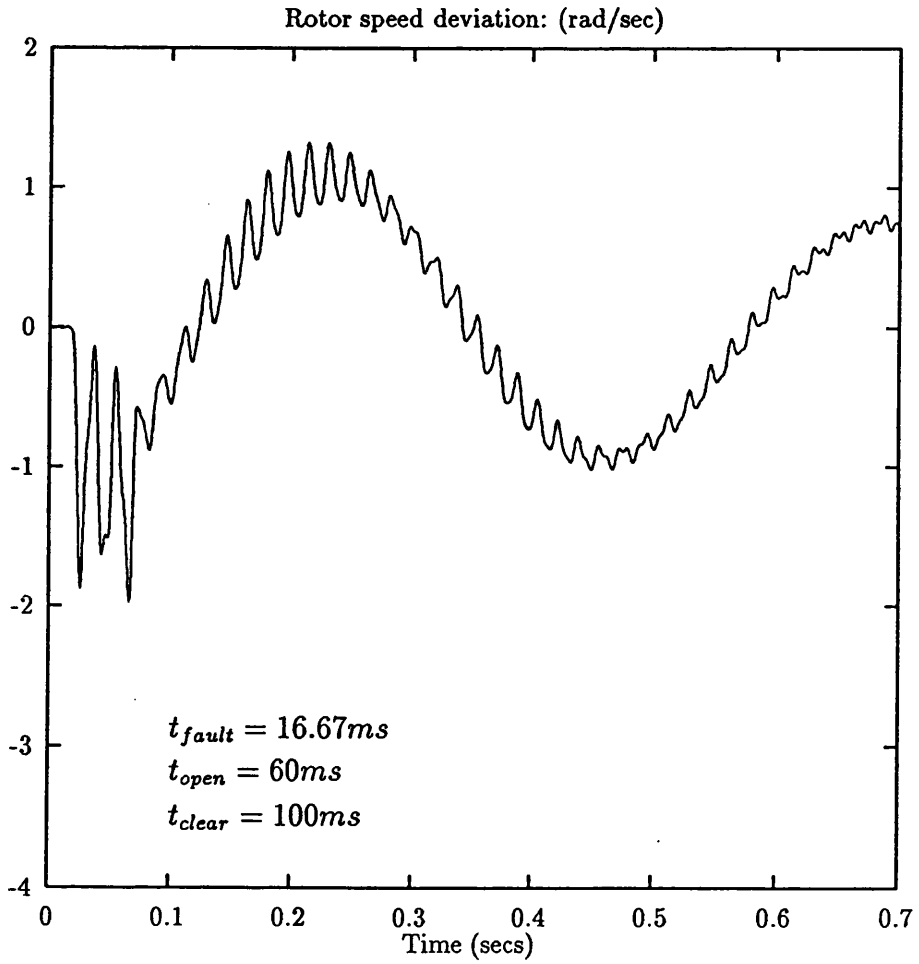

Figure 8.59: Actual source model: midpoint a-earth near voltage zero fault, rotor speed deviations.

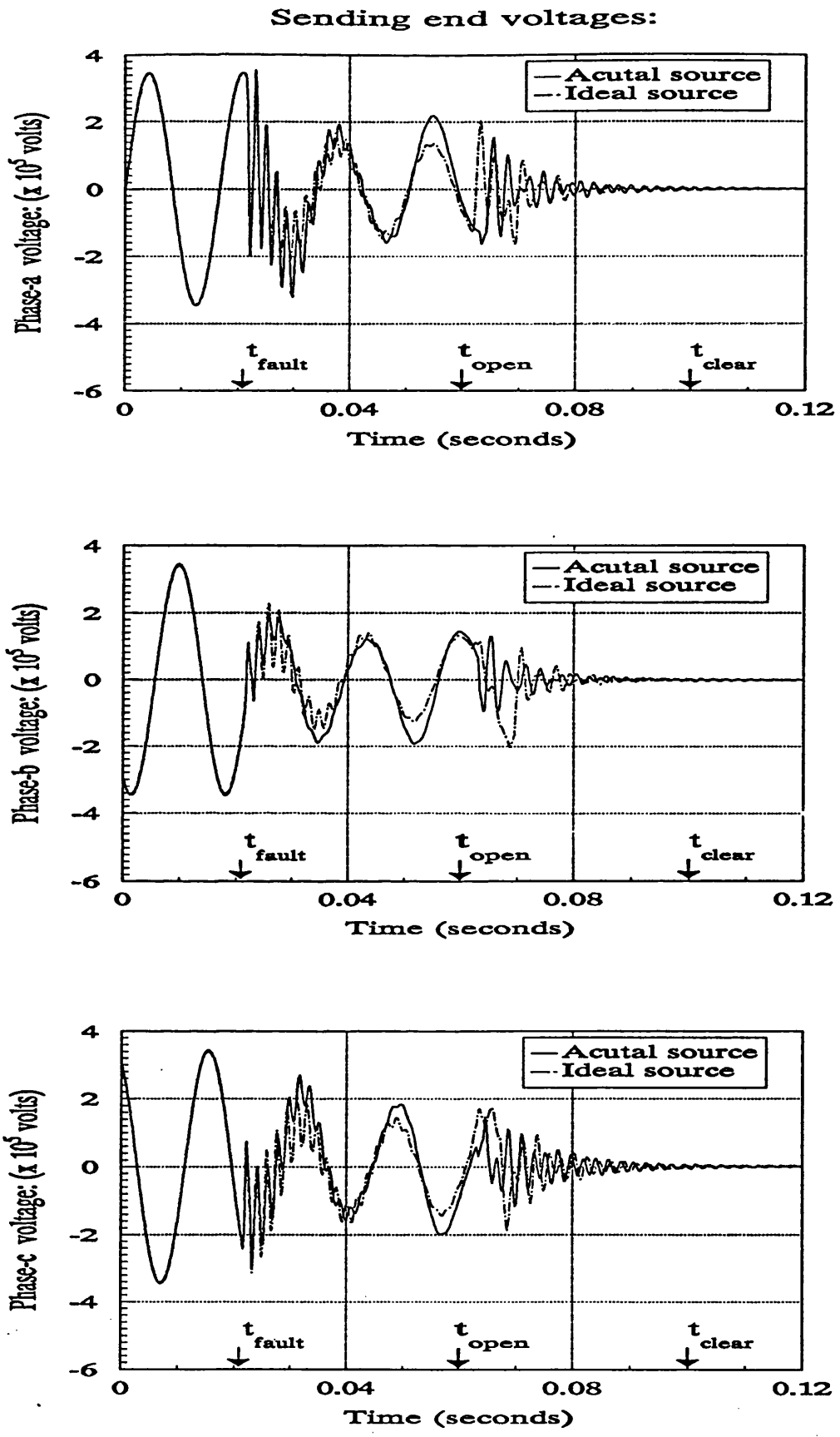

Figure 8.60: Actual and ideal source model comparison: midpoint three phase to earth fault.

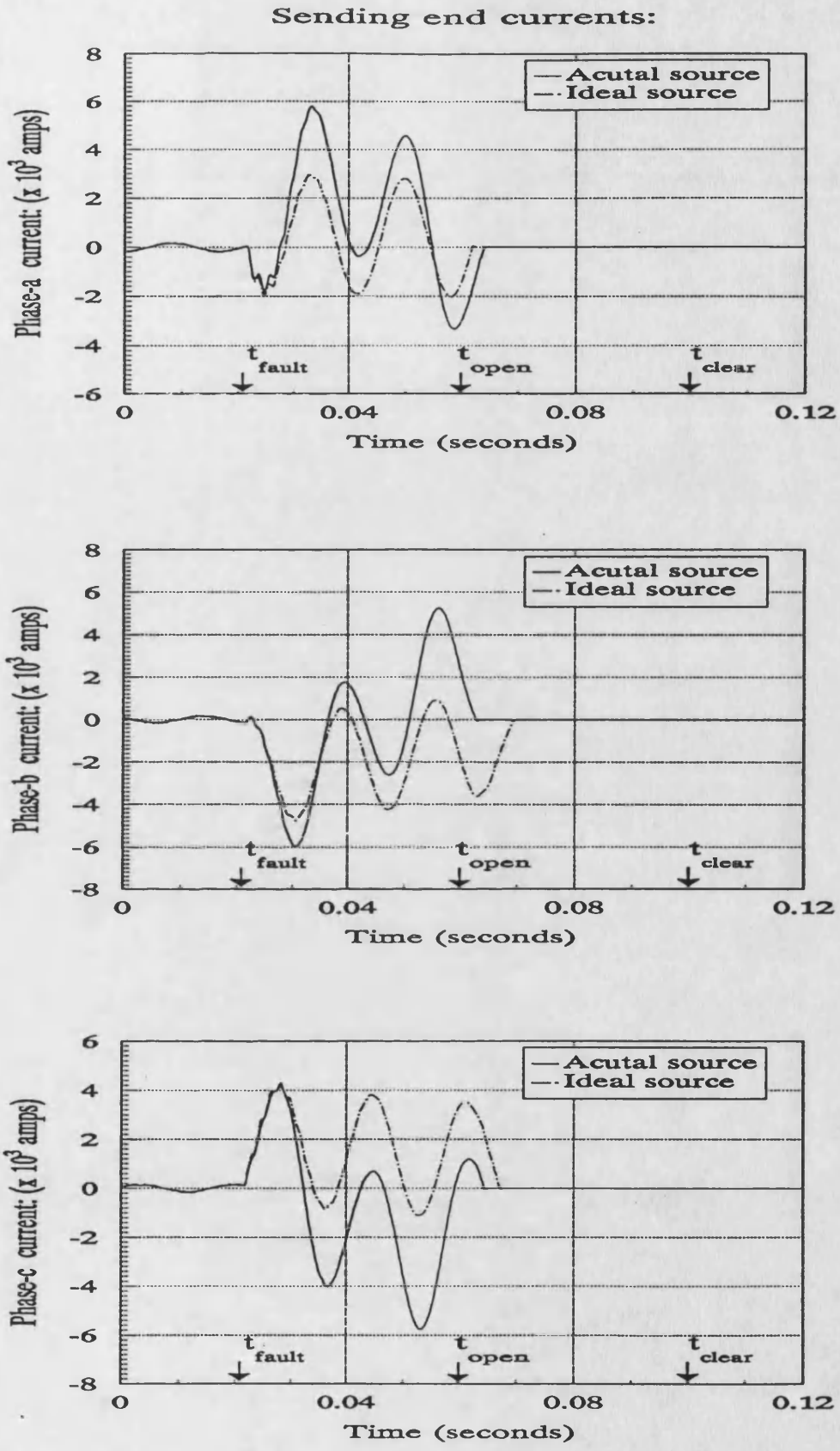

Figure 8.61: Actual and ideal source model comparison: midpoint three phase to earth fault.

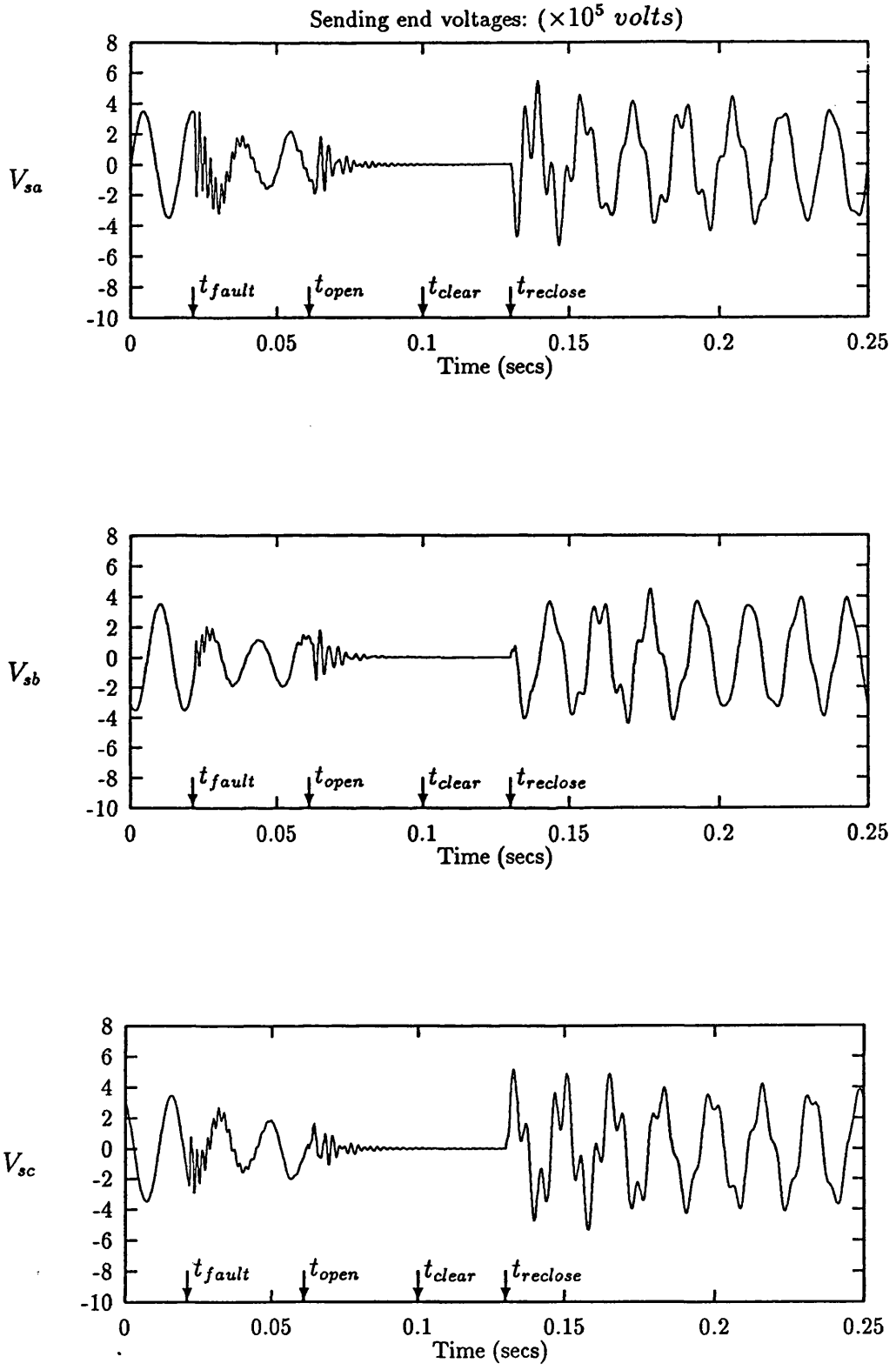

Figure 8.62: Actual source model: autoreclosure sequence for midpoint three phase to earth fault, breaker poles at both ends opened sequentially.

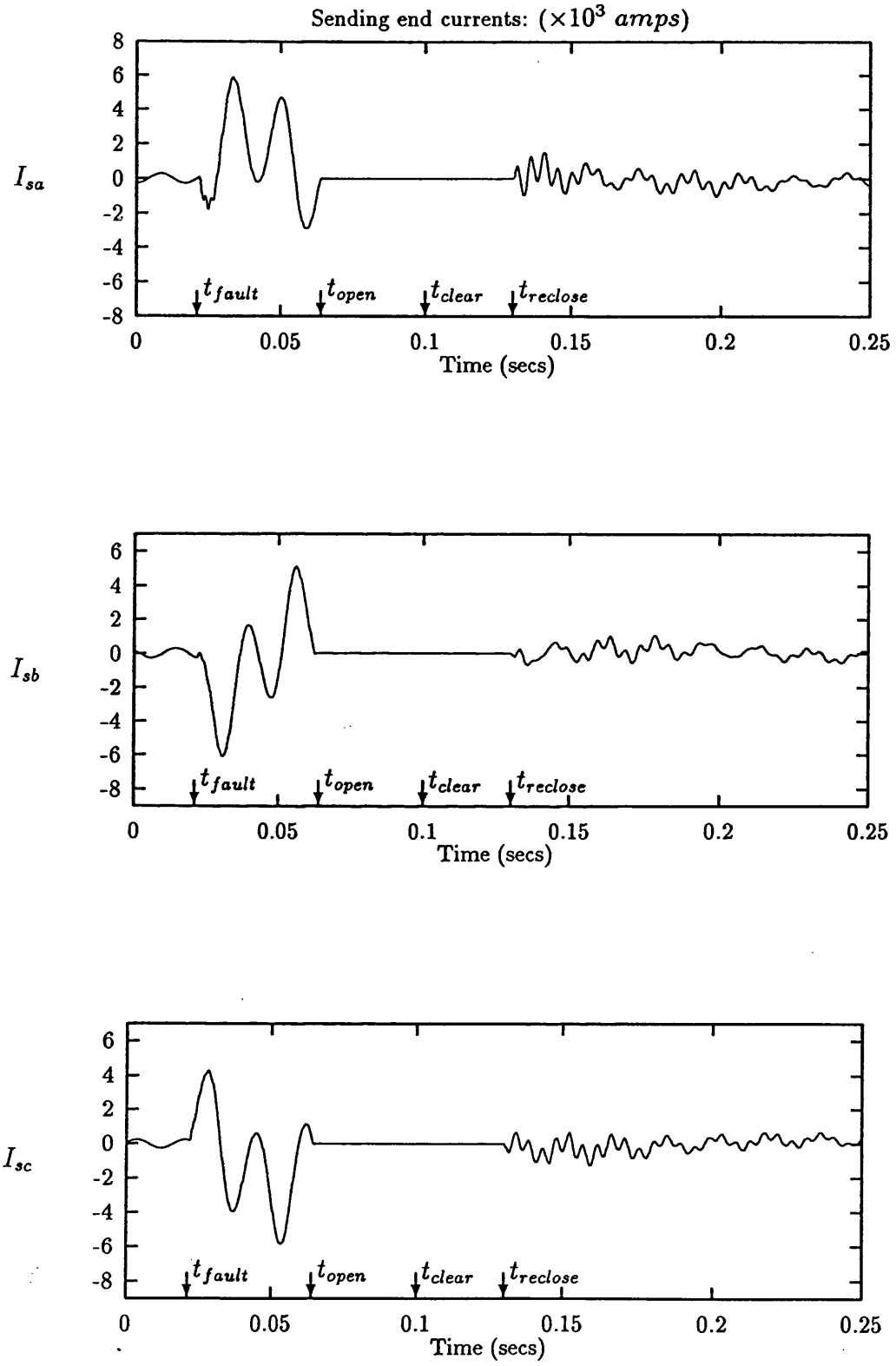

Figure 8.63: Actual source model: autoreclosure sequence for midpoint three phase to earth fault, breaker poles at both ends opened sequentially.

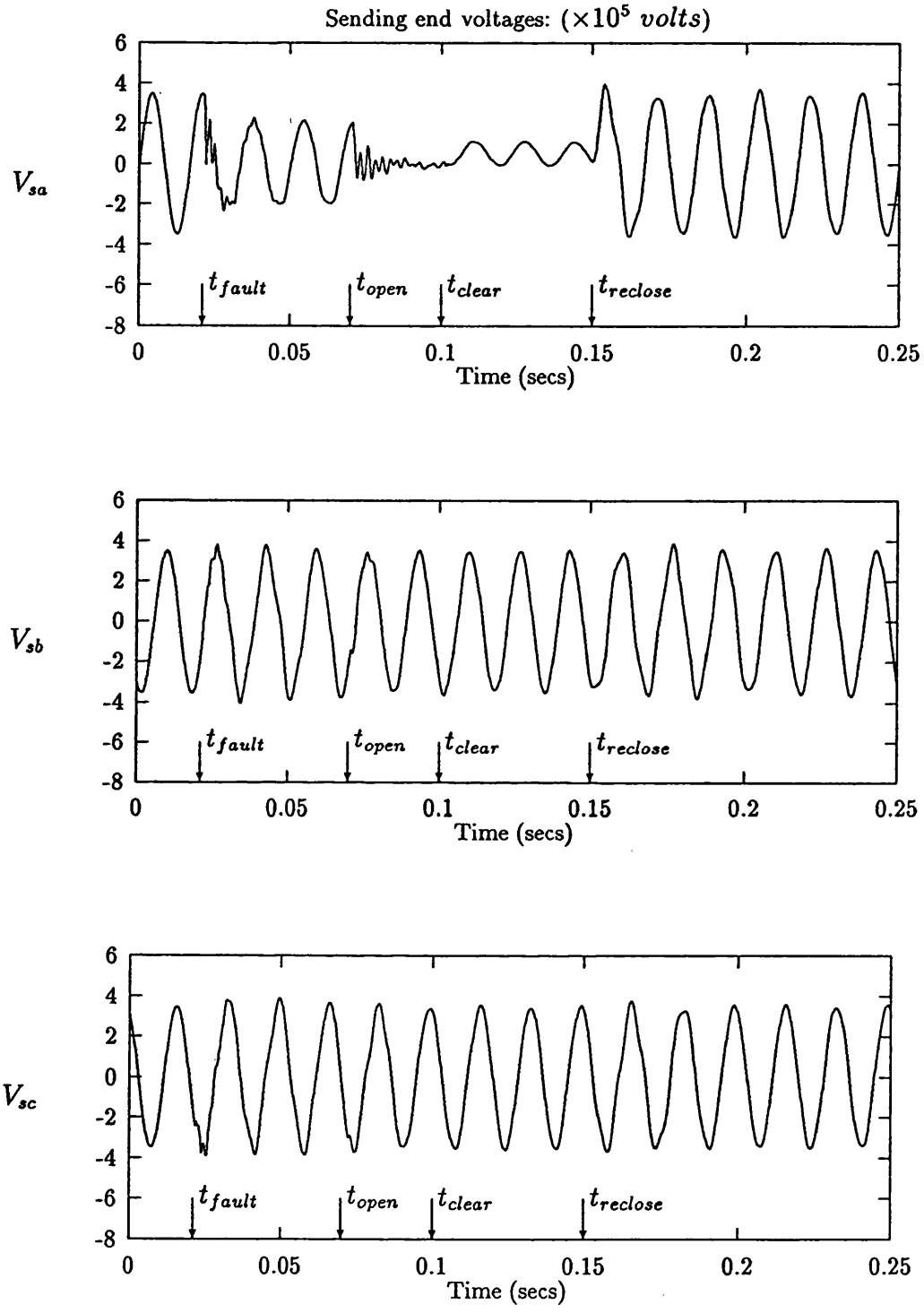

Figure 8.64: Actual source model: single pole autoreclosure sequence for midpoint phase-a to earth fault, breaker poles at both ends opened sequentially.

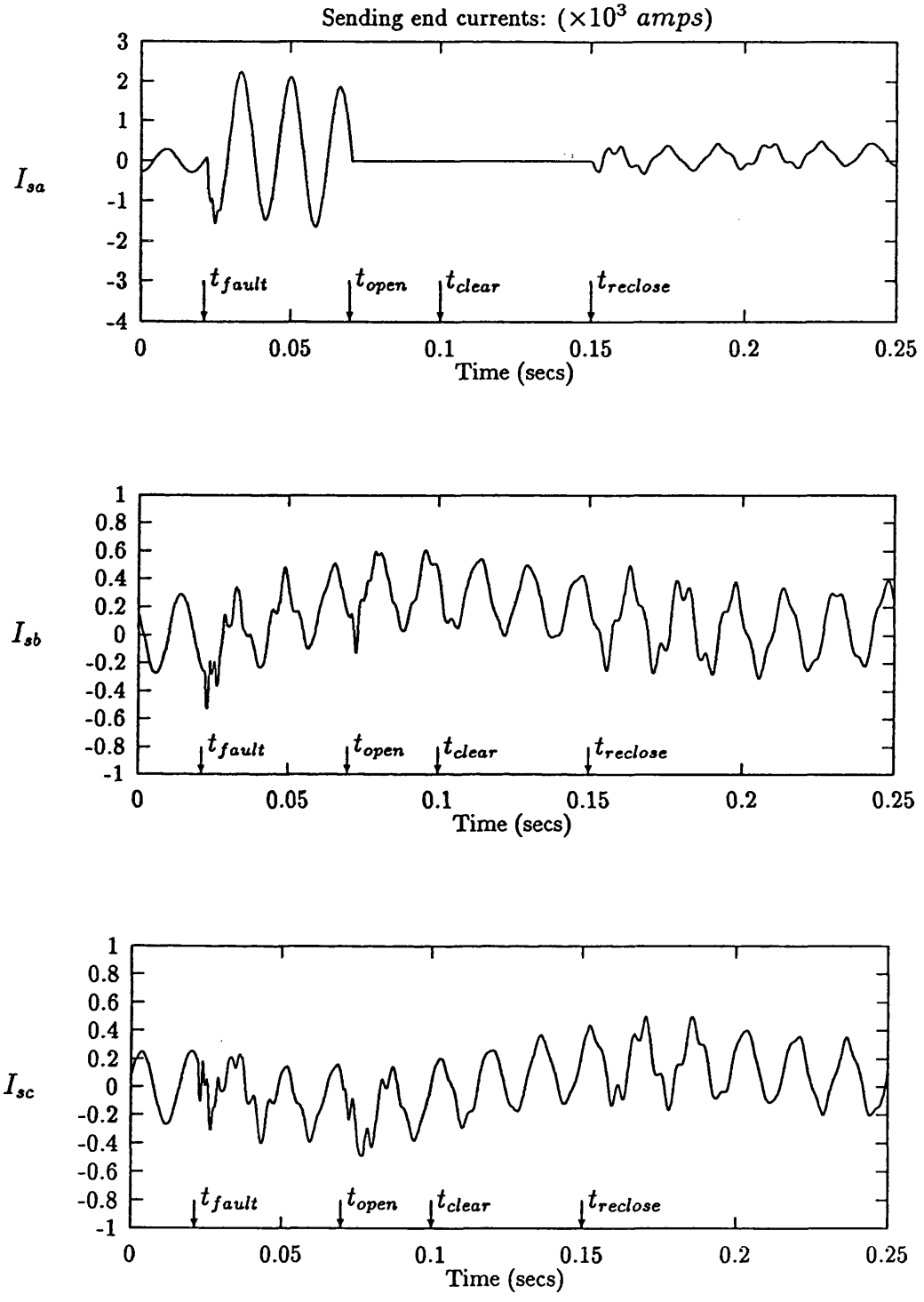

Figure 8.65: Actual source model: single pole autoreclosure sequence for midpoint phase-a to earth fault, breaker poles at both ends opened sequentially.

Chapter 9

Conclusions

Based on the time convolutional and the frequency-domain simulation techniques, optimised simulation codes for the electromagnetic transient simulation of fault generated autoreclosure sequence on EHV transmission systems have been developed and investigated. Algorithms for modelling both the electromagnetic and the electromechanical power system fault transients have also been developed based on the EMTP simulation program.

Time-convolutional technique

It has been found that the accuracy of the time-convolutional technique depends very much on the number of coefficients of the impulse response functions used. Erroneous results are obtained if the number of coefficients are chosen such that the impulse response function is truncated at a point in time where a significant amount of energy is still left in the impulse response. Some system impulse response functions used for the calculation of current transients, converge towards zero very slowly. In some cases, the impulse responses even start to diverge

towards the end. Difficulties were found for the time-convolutional technique to achieve the required accuracy, particularly for currents. An extensive series of tests have shown that a very large number of coefficients are required for the currents to achieve a reasonable degree of accuracy. As a result, in terms of both accuracy and computational time, the time convolutional technique has no advantage at all as compared to the frequency domain technique.

Frequency-domain technique

For the frequency domain simulation technique, the simulation waveforms obtained from the newly developed simulator are effectively identical to those generated from the traditional mainframe simulators of Johns and Aggarwal [1, 2]. Table 8.3 shows the computational time measured using a single Intel *i*860 processor. Assuming that one processor is used to calculate the transient of one variable, the worst computational time to obtain the sending end voltage, V_{sa}, for opening the circuit breaker poles is $15ms$ for each circuit change. It has been found that from a practical point of view, for three phase autoreclosure, it is only possible to simulate the electromagnetic transients and test a relay interactively in real time if all the circuit breaker poles are opened simultaneously. However, this would not be the case in practice. This is so because it would cause current choppings and enormous voltage transients would be produced. In practice, the circuit breaker poles are opened at current zeroes of the respective phases. This means that the breaker poles are opened at different times. The worst time interval between the opening of the first circuit breaker pole to the opening of the last circuit breaker pole may be less than $5ms$. Thus, after the opening of the first circuit breaker pole, there may be, for the worst case, another 5 subsequent circuit changes for a three phase system in a very short time period. The time

interval between each subsequent circuit changes can be less than $1ms$. The computational time required to calculate each variable of interest for any subsequent circuit change is also $15ms$.

It is worth noting that the frequency domain technique requires the calculation of the whole set of simulation samples before the first useful new transient output is obtained. This means that for each circuit change, the first new input sample, which can be used by the relay, can only be obtained after $15ms$ of transient computation. It should be recalled that there is around $40ms$ available for the simulator before the first circuit breaker pole opening. Thus, within this $40ms$, the simulator is able to calculate the first two breaker pole opening transients in real time. At this point, it can be seen that if the scheme employed is single pole autoreclosure, then it is possible for the relays to be tested interactively in real time with sequential pole operations. However, if three phase autoreclosure scheme is employed, the simulator is unable to calculate the remaining four subsequent breaker pole opening transients in real time. Since the time step length in real time is $250\mu s$, it can be seen that by neglecting the communication overhead amongst the processors, the simulator is about 56 times [1] slower than real time for each variable of interest [2] for each circuit change. This means that it is about 180 times [3] slower than real time all together for the four subsequent four circuit changes.

[1] Taking into account that there is about a $1ms$ time interval between each circuit change and neglecting the communication overhead among the processors, the ratio of the computational time over the real time is given by: $\{(15ms - 1ms)/250\mu s\} = 56$.

[2] There are altogether 18 such variables. It has been assumed that one processor is used to calculate the fault transient of each variable. If the communication overhead is also taken into account, then the above ratio will be greater than 56.

[3] It should be noted that the first two circuit changes require $30ms$ of transient computation. Thus, there is still $12ms$ available for the simulator to calculate the third circuit change. Considering the third circuit change alone, the simulator is about $\{(15ms - 12ms)/250\mu s\} = 12$ times slower than real time. Together with the remaining three subsequent circuit changes, the simulator is about $(12 + 56 \times 3) = 180$ times slower than real time.

The nonsimultaneity condition as discussed above can likewise occur on reclosure to produce sequential pole closure. It can thus be seen that for three phase autoreclosure scheme, the frequency domain technique is too far away from achieving the 'real time' requirement even with the highly optimised simulation codes running under one of the world's fastest microprocessors, the Number-Smasher, *i*860. It may require a large number of Intel *i*860 processors, which is physically and economically not viable, to achieve the real time requirement.

Based on the above judgements, it is concluded that for three phase autoreclosure, the frequency-domain technique is not suitable for power system fault transient simulation for the purposes of testing the relays interactively in real time. However, for single pole autoreclosure, it is possible for the technique to be used to test the relays interactively in real time.

EMTP technique

Some preliminary studies have shown that the EMTP technique can overcome the drawbacks which are suffered by the time-convolutional and the frequency-domain simulation techniques. Firstly, it is a time-domain technique. This means that it is not necessary for the data to be transformed into and out of the frequency-domain for each circuit change which occurs in the time-domain. Furthermore, as compared to the frequency domain technique, it is not necessary to calculate the whole set of the simulated samples before the first new output sample is obtained. In addition, in contrast to the large number of coefficients required for the time-convolutional technique to achieve reasonable accuracy, another benefit from using the EMTP is that each new output sample is derived from a few already known history terms. More importantly, it can also accurately incorporate

the frequency dependence of the system parameters by synthesising an equivalent system network in the time-domain.

Algorithms for modelling both the electromagnetic and the electromechanical power system fault transients have been developed based on the EMTP technique. The waveforms obtained from the newly developed simulators match closely to those obtained from the mainframe simulators by Johns and Aggarwal. It has been found that a simplified source model does not produce the same degree of dc offset as that produced by the actual generator model. For faults which occur near zero voltage, complete offsetting of the current waveforms can occur and significant differences are observed when compared to those obtained using the simplified source models. For faults which occur near the peak of the prefault voltage, the waveforms do not differ significantly from those observed using simplified source models based upon subtransient values. Furthermore, it should be mentioned that simplified source model cannot produce the low frequency electromechanical transient phenomenon, which is clearly shown when the actual generator model is used.

It should be mentioned that the simulating codes have not been optimized and are currently running in a serial mode. To achieve the 'real time' requirement, a great deal of work is still needed. For example, the simulating codes must be optimized and parallel processing algorithms should be properly designed with great care.

Chapter 10

Future Work

The power system fault transient modelling based on the EMTP modelling technique has been successfully developed. The results obtained have shown that this technique can overcome the drawbacks of the time convolutional and the frequency domain modelling techniques. It is therefore worthwhile to further develop and investigate the transient modelling based on this technique. It is proposed that the future work falls into the following categories:

1. From the point of realization of real time simulation, the EMTP program, for the time being, has two major weaknesses. Firstly, the simulator has been implemented and running in a serial mode. The simulating code has been written in a very inefficient way. In this respect, it is suggested that massive amount of effort must be devoted to modify and optimize the simulating code. Multiprocessing techniques should be suitably developed in order to bring the simulation to real time requirement as mentioned in Chapter 1.1 and 2.4.

2. A suitable algorithm, which is able to select the outputs from the simulators should be developed so that during the prefault clearance period,

the outputs from the electromagnetic program are selected and, during the postfault clearance period, the outputs from the electromechanical program are selected. For example, one processor could run the electromagnetic transient program and another could run the electromechanical program.

3. A single machine system has been modelled for the electromechanical transient studies. It may be desirable to incorporate a multimachine model and study the low frequency effects on the overall system.

4. It is suggested that a machine model and a generator/transformer model with frequency variant parameters should be incorporated.

5. It has been assumed that the arc resistance of the circuit breaker is ideal, i.e., the resistance jumps from zero to infinity at the instant of opening the breaker pole and vice versa. This may be quite impractical. It is suggested that the nonlinearity of the circuit breaker arc resistance should be modelled.

6. The modelling method should be extended to a double-circuit, multi-terminal system.

References

[1] JOHNS, A.T. & AGGARWAL, R.K.: 'Digital Simulation of faulted EHV transmission lines with particular reference to very high speed protection', *Proc. IEE*, 1976, **Vol–123**, pp.353–359.

[2] JOHNS, A.T. & AGGARWAL, R.K.: 'Digital simulation of fault autoreclosure sequences with particular reference to the performance evaluation of protection for EHV transmission lines', *Proc. IEE*, 1981, **Vol–128**, pp.183–195.

[3] BUDNER, A.: 'Introduction of frequency dependent line parameters into an electromagnetic transient program', *IEEE Trans.*, 1972, **PAS–91**, pp.85–91.

[4] HUMPAGE, W.D., WONG, K.P. & NGUYEN, T.T.: 'Time convolution and z-transform methods of electromagnetic transient analysis in power systems', *IEE Proc.*, 1980, **Vol–127**, pp.386–394.

[5] DOMMEL, H.W.: 'EMTP theory book', *Bonneville Power Administration Publication*, April 1987.

[6] Leuven EMTP Centre: 'EMTP rule book', Belgium, 1987.

[7] PETERSON, H.A.: 'Transients in power systems', *Dover, New York*, 1966.

[8] BERGERON L.: 'Du Coup de Belier en Hydraulique au Coup de Foudre en Electricite', Dunod, Paris, 1949. (English translation: 'Water hammer in

hydraulics and wave surges in electricity', ASME Committee, *Wiley, New York*, 1961.)

[9] BICKFORD, J.P. & DOEPEL, P.S.: 'Calculation of switching transients with particular reference to line energisation', *Proc. Inst. Elect. Eng.*, 1967, **Vol–114**, pp.465–477.

[10] DOMMEL, H.W.: 'Digital computer solution of electromagnetic transients in single- and multi-phase networks', *IEEE Trans.*, 1969, **PAS–88**, pp.388–399.

[11] DOMMEL, H.W., & MEYER, W.C.: 'Computation of Electromagnetic Transients', *Proc. IEEE*, 1974, **Vol–62**, No.7, pp.983–993.

[12] BATTISSON, M.J. *et al.*: 'Calculation of switching phenomena in power systems', *Proc. Inst. Elect. Eng.*, 1967, **Vol–114**, pp.478–486.

[13] BICKFORD, J.P., MULLINEUX, N. & REED, J.R.: 'Computation of power system transients', *IEE Monograph*, 1976, p.75.

[14] AMENTANI, A.: 'The application of the fast Fourier Transform to electrical transient phenomena', *Int. J. Elect. Eng. Educ.*, 1973, **Vol–10**, pp.277–287.

[15] AMENTANI, A.: 'A highly efficient method for calculating transmission line transients', *IEEE Trans.*, 1976, **PAS–95**, pp.1545–1551.

[16] HUMPAGE, W.D., WONG, K.P. & NGUYEN, T.T.: 'Impulse sampling sequence in time convolution electromagnetic transient analysis in power systems', *Electric Power System Research*, 1981, pp.13–20.

[17] HUMPAGE, W.D., WONG, K.P. & NGUYEN, T.T.: 'Digital filter model of power transmission line', *Electric Power & Energy Systems*, 1981, **Vol–3**, No.4, pp.197–207.

[18] UMOTO, J. & HARA. T.: 'A new digital analysis of surge performance in electric power networks using the convolution integral', *Elec. Eng. Japan*, 1971, 91, (3), pp.48–57.

[19] CARSON, J.R.: 'Wave propagation in overhead wires with ground return', *Bell System Technical Journal*, 1926, 5, pp.539–554.

[20] Perihelion Software Ltd., 'The Helios operating system', *Prentice-Hall*, 1989.

[21] Inmos Ltd, 'Transputer Reference Manual', *Prentice Hall*, 1988.

[22] Intel Corporation, 'Microprocessors', **Vol–II**, 1990.

[23] Intel Corporation, 'i860 Microprocessor family, programmer's reference manual', 1991.

[24] FRIED, S.S.: 'Personal supercomputing with the Intel i860', *PC Magazine, BYTE*, January 1991, pp.347–364.

[25] WEDEPOHL, L.M.: 'Application of matrix methods to the solution of travelling-wave phenomena in polyphase systems', *Proc. IEE*, 1963, **Vol–110**, pp.2200–2212.

[26] MAGNUSSON, P.C.: 'Travelling waves on multi-conductor open-wire lines: A numerical survey of the effects of frequency dependence of modal composition', *IEEE Trans.*, 1973, **PAS–92**, pp.999–1008.

[27] WASLEY, R.G. & SELVAVINAYAGAMOORTHY, S.: 'Approximate frequency response values for transmission line transient analysis', *IEEE Proc.*, 1974, **Vol–121**, pp.281–286.

[28] SNELSON, J.K.: 'Propagation of travelling waves on transmission lines – frequency-dependent parameters', *IEEE Trans.*, 1972, **PAS–91**, pp.85–91.

[29] MEYER, W.S. & DOMMEL, H.W.: 'Numerical modelling of frequency-dependent transmission line parameters in an electro-magnetic transient program', *IEEE Trans.*, 1974, **PAS–93**, pp.1401–1409.

[30] MARTI, J.: 'Accurate modelling of frequency-dependent transmission lines in electromagnetic transients simulations', *IEEE Trans.*, 1982, **PAS–101**, pp.147–155.

[31] ANDERSON, P.M. & FOUAD, A.A.: 'Power system control and stability', *The Iowa State University Press*, 1977.

[32] KIMBARK, E.W.: 'Power system stability', **Vol–3**, 'Synchronous machines', *Wiley, New York*, 1962.

[33] IEEE: 'IEEE standard dictionary of electrical and electronics terms', Second edition, *Wiley, New York*, 1977, p.700.

[34] IEEE Committee Report: 'Excitation system models for power system stability studies', *IEEE Trans.*, 1981, **PAS–100**, pp.494–509.

[35] IEEE Committee Report: 'Dynamic models for steam and hydro turbines in power system studies', *IEEE Trans.*, 1972, **PAS–91** pp.1904–1914.

[36] LOY, N.J.: 'An engineer's guide to FIR digital filters', *Prentice Hall*, 1988.

[37] DAY, J., MULLINEUX, N. & REED, J.R.: 'Developments in obtaining transient response using Fourier Transforms', *J. Elect. Eng. Educ.*, Part 3, 1965, **Vol–3**, pp.501–506.

[38] WEDEPOHL, L.M. & MOHAMED, S.E.T.: 'Multiconductor transmission lines', *Proc. IEE*, 1969, **Vol–116**, pp.1553–1563.

[39] BRIGHAM, E.O.: 'The fast Fourier transform', *Prentice-Hall*, 1974.

[40] Perihelion Software Ltd., 'The Helios Technical Guide: The CDL guide', *Distributed Software Ltd.*, 1990.

[41] JOHNS, A.T. & M.M.T. EI-KATEB: 'Developments in techniques for simulating faults in EHV transmission systems', *IEE Proc.*, 1978, **Vol–125**, pp.221–229.

[42] OWEN, R.E. & LEWISE, W.A.: 'Asymmetry characteristic of progressive short-circuit on large synchronous generators', *IEEE Trans.*, 1971, **PAS–90**, pp.587–596.

[43] BAXENDALE, W.: 'Power system dynamic stability studies', *M.Sc. Thesis*, University of Bath, 1972.

[44] BERRY, T.: *Private communication*, School of Mechanical Engineering, University of Bath, 1992.

[45] CARROLL, D.P. & NOZARI, F.: 'An efficient computer method for simulating transients on transmission lines with frequency dependent parameters', *IEEE Trans.*, 1975, **PAS–94**, No.4, pp.1167–1176.

[46] DALE, L.A.: 'Real time modelling of multimachine power systems', *Ph.D. Thesis*, University of Bath, 1985.

[47] FREY, W. & ALTHAMMER, P.: 'The calculation of electromagnetic transients on lines by means of a digital computer', *The Brown Boveri Review*, May/June, 1961, **Vol–48**, pp.344–355.

[48] HARBISON, S.P. and STEELE, G.L: 'C: A reference manual', Prentice-Hall, 1987.

[49] LAI, L.L., JOHNS, A.T. & DARUVALA, D.J.: 'Modelling of an arcing faults using the electromagnetic transients program (EMTP)', *Proc. UPEC.*, 1990, **Vol–2**, pp.675–678.

[50] RABINER, L.R. & GOLD, B.: 'Theory and application of digital signal processing', *Prentice-Hall*, 1975.

[51] SEMLYEN, A. & DABULEANU, A.: 'Fast and accurate switching transient calculations on transmission lines with ground return using recursive convolutions', *IEEE Trans.*, 1975, **PAS–94**, pp.561–571.

[52] The General Electric Company: 'Protective relays application guide', *GEC measurements*, 1975.

[53] WEEDY, B.M.: 'Electric power systems', Third edition, *Wiley, New York*, 1983.

Appendix A

Basic definitions for autoreclosure operations

A number of the terms defined below are illustrated in fig. 2.1 and fig. 2.2, which illustrate the sequence of events in a typical auto-reclose operation, where the circuit breaker makes one attempt at reclosure after tripping to clear a fault. Two conditions are shown; a successful reclosure in the event of the fault being transient, and an unsuccessful reclosure followed by lock-out of the circuit breaker if the fault is permanent.

Anti-pumping: A feature incorporated in the circuit breaker or reclosing scheme whereby, in the event of a permanent fault, repeated operations of the circuit breaker are prevented when the closing impulse lasts linger than the sum of the protective relay and circuit breaker operating times.

Arcing time: The time between the instant of separation of the circuit breaker contacts and the instant of extinction of the fault arc.

Closing impulse time: The time during which the closing contacts of the auto-reclose relay are made.

Closing time: The time from the energizing of the circuit breaker closing circuit to the making of the circuit breaker contacts.

Counting relay: A relay, usually of the electromagnetic type, with a ratchet mechanism which is driven forward one step each time its coil is energized. A contact is operated after a chosen number of steps and the mechanism may be manually or electrically reset.

Dead time (autoreclose relay): The time between the auto-reclose scheme being energized and the operation of the contacts which energize the circuit breaker closing circuit. On all but instantaneous or very high speed reclosing schemes, this time is virtually the same as the circuit breaker dead time.

Dead time (circuit breaker or system): The time between the fault arc being extinguished and the circuit breaker contacts remaking.

De-ionizing time: The time following the extinction of an overhead line fault arc necessary to ensure dispersion of ionized air so that the arc will not re-strike when the line is re-energized.

High speed reclosing scheme: A scheme whereby a circuit breaker is automatically reclosed within 1 second after a fault trip operation.

Lock-out: A feature of an auto-reclose scheme which, after tripping of the circuit breaker, prevents further automatic reclosing.

Multi-shot reclosing: An operating sequence providing more than one reclosing operation on a given fault before lock-out of the circuit breaker occurs.

Single-shot reclosing: An operating sequence providing only one reclosing operation, lock-out of the circuit breaker occurring on subsequent tripping.

Operation counter: A counter, usually of the electromagnetic cyclometer type, arranged to indicate the number of automatic operations, either closing or tripping, performed by a circuit breaker since its commissioning.

Opening time: The time between the energizing of the circuit breaker trip coil and the instant of separation of the contacts.

Operating time (circuit breaker): The time from the energizing of the trip coil until the fault arc is extinguished.

Operating time (protection): The time from the inception of fault to the closing of the tripping contacts. Where a separate tripping relay is employed, its operating time is include.

Reclaim time: The time following a successful closing operation, measured from the instant the autoreclose relay closing contacts make, which must elapse before the auto-reclose relay will initiate a reclosing sequence in the event of a further fault incident.

System disturbance time: The time between the inception of the fault and the circuit breaker contacts making on successful reclosing.

Appendix B

Derivations of system Y and Z matrices

B.1 System Y-matrix

Referring to fig. 4.1, the following circuit analyses can be carried out for a single conductor system:

$$V_s = E_s - Z_{ss} I_s \tag{B.1}$$

$$V_r = E_r - Z_{sr} I_r \tag{B.2}$$

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} E_{ff} \\ I_{fs} \end{bmatrix} \tag{B.3}$$

$$\begin{bmatrix} V_r \\ I_r \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} E_{ff} \\ I_{fr} \end{bmatrix} \tag{B.4}$$

$$I_f = I_{fs} + I_{fr} \tag{B.5}$$

where Z_{ss} and Z_{sr} are respectively the sending end source impedance and the receiving end source impedance. Elements A, B, C & D are defined in section 4.2.2

for a single conductor system. Similar forms of system equations can be obtained for a single circuit three phase system (fig. 4.2). The only difference is that for a three phase system, the voltages and currents are 3 element vectors (one per phase) and elements A, B, C & D are 3 by 3 submatrices. Detailed derivations of these ABCD matrices are given in Chapter 3.3.6. Manipulation of the two preceding equations yields the universal admittance matrix, Y, which relates the sending end, the receiving end and the fault point parameters:

$$\begin{bmatrix} I_f \\ I_s \\ I_r \end{bmatrix} = \begin{bmatrix} Y_A & Y_B & Y_C \\ Y_D & Y_E & Y_F \\ Y_G & Y_H & Y_I \end{bmatrix} \begin{bmatrix} E_{ff} \\ E_s \\ E_r \end{bmatrix} \tag{B.6}$$

Let Z_f be the fault path impedance matrix, the following two sets of Y-matrix constants can be obtained:

Neglecting fault path impedance

If $Z_f = 0$, then $E_{ff} = V_f$ and

$$\begin{aligned} Y_{A0} &= -(Z_{ss}D_1 + B_1)^{-1}(A_1 + Z_{ss}C_1) \\ &\quad -(Z_{sr}D_2 + B_2)^{-1}(A_2 + Z_{sr}C_2) && \text{(B.7)} \\ Y_{B0} &= (Z_{ss}D_1 + B_1)^{-1} && \text{(B.8)} \\ Y_{C0} &= (Z_{sr}D_2 + B_2)^{-1} && \text{(B.9)} \\ Y_{D0} &= -(Z_{ss}D_1 + B_1)^{-1} && \text{(B.10)} \\ Y_{E0} &= D_1(Z_{ss}D_1 + B_1)^{-1} && \text{(B.11)} \\ Y_{F0} &= 0 && \text{(B.12)} \\ Y_{G0} &= -(Z_{sr}D_2 + B_2)^{-1} && \text{(B.13)} \\ Y_{H0} &= 0 && \text{(B.14)} \\ Y_{I0} &= D_2(Z_{sr}D_2 + B_2)^{-1} && \text{(B.15)} \end{aligned}$$

Nonzero Z_f

If Z_f is nonzero, then $V_f = E_{ff} + Z_f I_f$ and the above values are modified as shown below:

$$Y_A = (U - Y_{A0}Z_f)^{-1}Y_{A0} \quad \text{(B.16)}$$

$$Y_B = (U - Y_{A0}Z_f)^{-1}Y_{B0} \quad \text{(B.17)}$$

$$Y_C = (U - Y_{A0}Z_f)^{-1}Y_{C0} \quad \text{(B.18)}$$

$$Y_D = (U - Y_{A0}Z_f)^{-1}Y_{D0} \quad \text{(B.19)}$$

$$Y_E = (U - Y_{A0}Z_f)^{-1}Y_{D0}Y_{B0}Z_f + Y_{E0} \quad \text{(B.20)}$$

$$Y_F = (U - Y_{A0}Z_f)^{-1}Y_{D0}Y_{C0}Z_f \quad \text{(B.21)}$$

$$Y_G = (U - Y_{A0}Z_f)^{-1}Y_{G0} \quad \text{(B.22)}$$

$$Y_H = (U - Y_{A0}Z_f)^{-1}Y_{G0}Y_{B0}Z_f \quad \text{(B.23)}$$

$$Y_I = (U - Y_{A0}Z_f)^{-1}Y_{G0}Y_{C0}Z_f + Y_{I0} \quad \text{(B.24)}$$

where U stands for unit matrix and 0 stands for zero matrix.

B.2 Fault path impedance matrix

Considering a typical double end fed three phase system as shown in fig. 4.2, let the three phases are referred to as ϕ_1, ϕ_2 and ϕ_3, and 0 as ground. Then Y_{10} is the fault admittance from ϕ_1 to ground, Y_{12} is the fault admittance from ϕ_1 to ϕ_2 and so on. The fault path admittance matrix can now be defined as:

$$Y_f = \begin{bmatrix} Y_{10}+Y_{12}+Y_{13} & -Y_{12} & -Y_{13} \\ -Y_{12} & Y_{20}+Y_{21}+Y_{23} & -Y_{23} \\ -Y_{13} & -Y_{23} & Y_{30}+Y_{31}+Y_{32} \end{bmatrix} \quad \text{(B.25)}$$

$$Z_f = Y_f^{-1} \quad \text{(B.26)}$$

B.3 System Z-matrix

The system Z-matrix is obtained by inverting the system Y-matrix, i.e.,

$$\begin{bmatrix} E_{ff} \\ E_s \\ E_r \end{bmatrix} = \begin{bmatrix} Z_A & Z_B & Z_C \\ Z_D & Z_E & Z_F \\ Z_G & Z_H & Z_I \end{bmatrix} \begin{bmatrix} I_f \\ I_s \\ I_r \end{bmatrix} \tag{B.27}$$

Appendix C

Fast Fourier transform

The Discrete Fourier Transform (DFT) converts samples of a time-domain signal to samples of the different signal frequencies and their amplitudes. To perform the conversion, the DFT calculation requires n^2 complex multiplications and additions; where n equals the number of samples and is usually a power of two.

The Fast Fourier Transform (FFT) [39] is an elegant implementation of the DFT, which reduces the number of complex multiplications and additions required to $n \log_2 n$. That is more than 100 times fewer for a 1024-point FFT. The FFT algorithm also has the advantage of not requiring data storage beyond the original data samples.

To accomplish the transform, a DFT adds combinations of the input data samples after multiplying those inputs with weighting factors. Each result in the frequency domain comes from a weighted sum of all data samples. The weighting factors W are called *twiddles*, and are complex cosine/sine values for each particular frequency.

The DFT $X(k)$ of a time-domain signal $x(t)$ is defined by:

$$\begin{aligned} x(n) &= \text{time-domain samples, } n = 0, 1, \cdots, N-1 \\ X(k) &= \text{the DFT of } x(n),\ k = 0, 1, \cdots, N-1 \\ &= \sum_{n=0}^{N-1} x(n) \times W^{nk}, \text{ and } W^{nk} = e^{-j2\pi nk/N} \\ &= \sum_{n=0}^{N-1} x(n) \times \{\cos(2\pi nk/N) - j\sin(2\pi nk/N)\} \end{aligned}$$

The $(N-1)$ complex additions and $(N-1)$ complex multiplications required for each $X(k)$ make the DFT into an $order(N^2)$ computation.

Fortunately, the FFT decomposes this to an $order(N \times \log_2 N)$ algorithm by splitting the N-sum into units of 2-sums. Thus, a DFT of 1024 input points requiring about two million operations reduces to an FFT requiring about fifteen thousand operations. The 2-sum units are called *butterflies* because they produce two output values from two inputs, with the butterfly-shaped data flow shown in fig. C.1. The butterfly calculations are executed in stages, with $\log_2 N$ stages and $N/2$ butterflies per stage.

As indicated in fig. C.1, the subdivision, or decimation, of the N-sum into butterflies can be done via two different methods: DIF (Decimation in Frequency) or DIT (Decimation in Time). The methods differ in the ordering of twiddles and the form of the butterfly arithmetic, but they yield the same answer. They are based on different mathematical derivations of the FFT: DIT results from recursively splitting the input time-domain samples into an even-indexed group and an odd-indexed group, while DIF comes from splitting the DFT output frequency-domain points into odd/even groups.

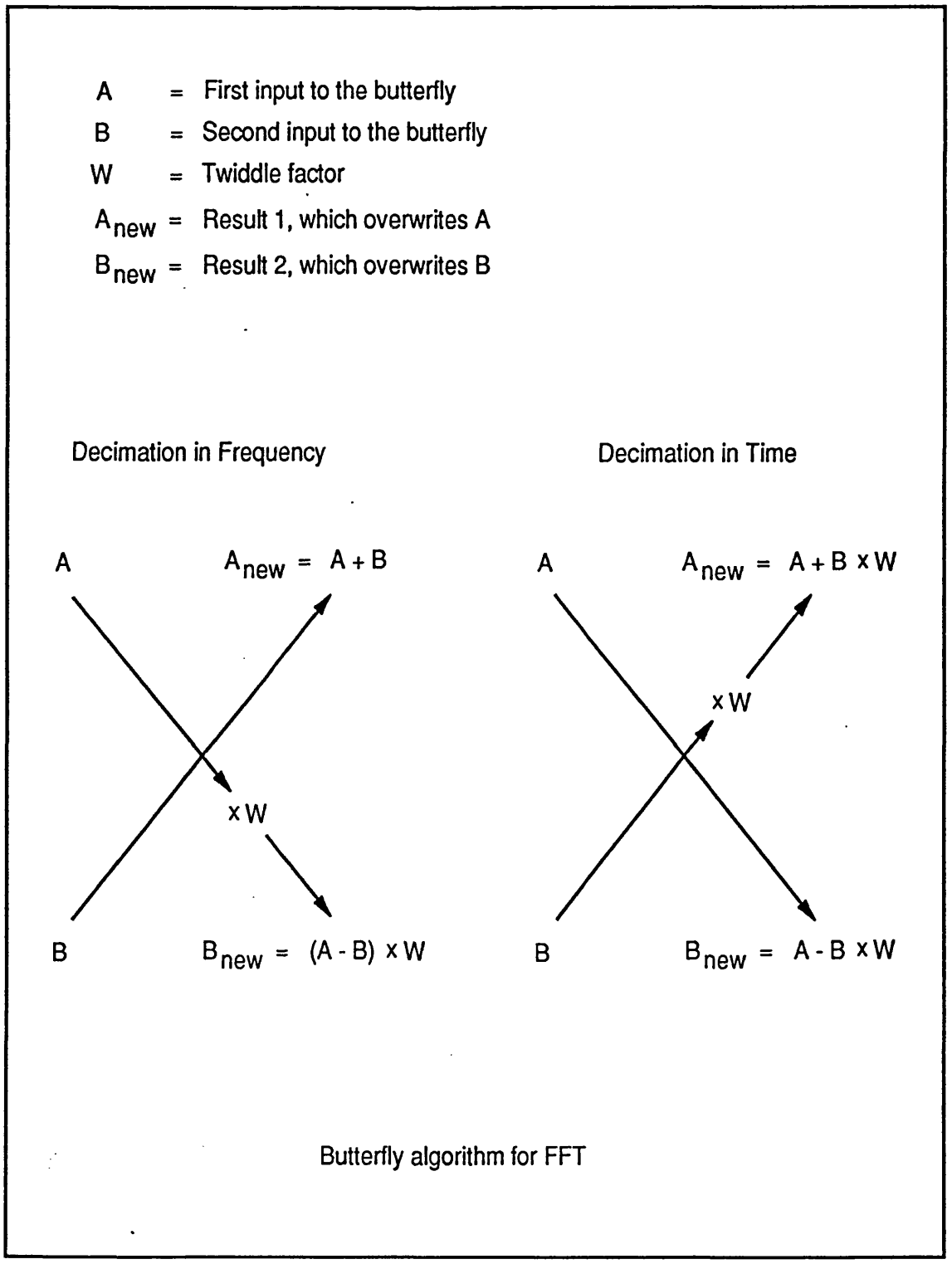

Figure C.1: Butterfly algorithm for FFT

Appendix D

Complex number library

Some basic complex number calculation routines such as addition, subtraction, multiplication, modulus, etc, are stored in the complex number library. These functions take pointers to complex structures as arguments. All functions are written such that the result pointer may be one of the source operands. In general if the result is complex number then the first argument pointer is returned as the result pointer. The simple functions are defined as macros. This may be inefficient if the macro arguments are expressions. The complex number structure is defined as shown below.

```
/* complex structure defined in "complex.h" */
typedef struct complex
{  float re;
   float im;
} complex;
```

In order to manipulate operations using complex numbers, "complex.h" should be included. The complex structures must be allocated with suitable memory by

the calling program. The use of each function routine declared in "complex.h" is explained in this appendix.

- **ARG:**

```
#include "complex.h"
float arg(complex *a)
```

The function **arg** calculates the argument of a *complex number* which is given by the argument *a*. The resulting value is returned. It is measured in radians.

- **CONJUGATE:**

```
#include "complex.h"
complex *conjugate(complex *a, complex *b)
```

The function **conjugate** conjugates a *complex number* which is given by the second argument *b*. The resulting value is stored in the first argument *a*. A pointer to the resulting *complex number* is returned.

- **DIVCOMPL:**

```
#include "complex.h"
complex *divcompl(complex *a, complex *b, complex *c)
```

The function **divcompl** divides a *complex number b* with a *complex number c* according to the formula $a = (b/c)$. The resulting value is stored in *complex number a*. A pointer to the resulting value is returned.

- **FPRINTF_COMPL:**

```
#include "complex.h"
complex *fprintf_compl(FILE *outptr, complex *a)
```

The function **fprintf_compl** prints a *complex number* a into an output file which is given by the first argument *outptr*.

- **MOD_COMPL:**

```
#include "complex.h"
float mod_compl(complex *a)
```

The function **mod_compl** calculates the modulus of a *complex number* which is given by the argument *a*. The resulting value is returned.

- **MOD_SQ_COMPL:**

```
#include "complex.h"
float mod_sq_compl(complex *a)
```

The function **mod_sq_compl** calculates the square of the modulus of a *complex number* which is given by the argument *a*. The resulting value is returned.

- **MOVECOMPL:**

```
#include "complex.h"
complex *movecompl(complex *a, complex *b)
```

The function **movecompl** copies the contents of a *complex number*, which is given by the second argument *b*, to another *complex number* which is given by the first argument *a*. A pointer to the moved complex number is returned.

- **MULTCOMPL:**

```
#include "complex.h"
complex *multcompl(complex *a, complex *b, complex *c)
```

The function **multcompl** multiplies a *complex number b* with a *complex number c* according to the formula $a = b \times c$. The resulting value is stored in *complex number a.* A pointer to the resulting value is returned.

- **NEGMULTCOMPL:**

```
#include "complex.h"
complex *negmultcompl(complex *a, complex *b, complex *c)
```

The function **negmultcompl** multiplies a *complex number b* with a *complex number c* according to the formula $a = -(b \times c)$. The resulting value is stored in *complex number a.* A pointer to the resulting value is returned.

- **RECIPCOMPL:**

```
#include "complex.h"
complex *recipcompl(complex *a, complex *b)
```

The function **recipcompl** reciprocates a *complex number b* according to the formula $a = \frac{1}{b}$. The resulting value is stored in *complex number a.* A pointer to the resulting value is returned.

- **SETCOMPL:**

```
#include "complex.h"
complex *setcompl(complex *a, float r, float i)
```

The function **setcompl** sets two *floating point numbers r* and *i* into a *complex number.* The resulting complex number is stored in the first argument *a.* The second argument *r* gives the real part of the complex number while the third argument *i* gives the imaginary part.

- **SUBMULTCOMPL:**

```
#include "complex.h"
complex *submultcompl(complex *a, complex *b, complex *c)
```

The function **submultcompl** multiplies and subtracts a *complex number* b with a *complex number* c according to the formula $a = a - (b \times c)$. The resulting value is stored in *complex number* a. A pointer to the resulting value is returned.